

# INFORMATION GENEALOGY: MODELING IDEA ORIGINS AND FLOWS IN TEXT

A Dissertation

Presented to the Faculty of the Graduate School  
of Cornell University

in Partial Fulfillment of the Requirements for the Degree of  
Doctor of Philosophy

by

Benyah Shaparenko

February 2010

© 2010 Benyah Shaparenko

ALL RIGHTS RESERVED

# INFORMATION GENEALOGY: MODELING IDEA ORIGINS AND FLOWS IN TEXT

Benyah Shaparenko, Ph.D.

Cornell University 2010

One goal of text mining is to provide automatic methods to help people grasp the key ideas in ever-increasing document collections. Often these text corpora accumulate incrementally over time by a self-referential process as documents propose new ideas, build on or refute existing ideas, or draw connections between different existing ideas, and so on. Such corpora are pervasive, including email, news articles, blogs, and research publications. Search engines are effective for retrieving individual documents from such corpora, but they do not typically provide information about the structure of the corpora and how their ideas developed over time.

We propose a set of tasks, which we call information genealogy, which seek to analyze and summarize a document collection's development over time in terms of its ideas. These methods focus on helping people grasp the document collection as a whole. Specifically, we address the following tasks: What is each document's (interesting) original contribution of ideas to the corpus? How do ideas flow from one document to another? What are the most important, influential documents and ideas?

We develop methods grounded in probability and statistics, specifically based on generative mixture models for document language modeling. Consequently, unlike heuristic approaches, these methods are both extensible and readily analyzable. In addition, the input for these methods consists of only the text and temporal ordering of the documents, not any hyperlink information. Exclusively using document text in an unsupervised setting allows these methods to apply in many domains. We evaluate these

methods on both synthetically-generated and actual research publications. In general, these methods outperform heuristic baseline methods based on text similarity alone.

## **BIOGRAPHICAL SKETCH**

Benyah Shaparenko was born in Blacksburg, VA and grew up in Myerstown, PA. He graduated from Dalet School in 1999 and went on to complete a Bachelor of Science in computer science from the Pennsylvania State University, University Park, PA in 2003, graduating with highest distinction and a minor in mathematics. He then went to Cornell University, Ithaca, NY to work toward this Ph.D., with a Master of Science in computer science from Cornell in 2008 along the way.

To my family.

## ACKNOWLEDGEMENTS

I would like to thank the many people who provided comments, encouragement, and support to help me complete this work.

Most importantly, my adviser, Thorsten Joachims, has guided me through all phases of this research with crucial insights in deriving methods, designing experiments, and presenting the work clearly. Through this process, he exhibited great patience and confidence, while steadily guiding and encouraging me toward a thesis. For this, I owe him a great debt of gratitude.

I would also like to thank the other members of my committee, Paul Ginsparg for insightful discussions and providing data from the Physics Arxiv, John Hopcroft for interesting discussions as I was beginning my thesis research, and Robin McNeal for allowing me to fulfill a personal interest in Chinese.

There are many individuals with whom I have discussed this work or collaborated. The early ideas that led to this work were based on discussions with Rich Caruana and Johannes Gehrke. Without the space to go into details, I would just like to thank the following non-committee faculty and students who have provided feedback and discussion about this work: Erick Breck, Cristian Danescu-Niculescu, Yookyung Jo, Nikos Karampatziakis, Art Munson, Alex Niculescu-Mizil, Filip Radlinski, Dan Sheldon, Adam Siepel, Ainur Yessenalina, Chun-Nam Yu, Yisong Yue, and the Machine Learning Discussion Group.

I would also like to thank my family for emotional support and encouragement, my parents Raymond and Alice Shaparenko, and my sisters and brother, Bithyah, Berayah, Barukyah, and Binayah. In Upson Hall and around Cornell, many people were great to work and have fun with, not only providing necessary distractions at times but also providing greater encouragement while I worked on my thesis, including Ainur, Alex, Art, Bernard, Chun-Nam, Dan, Daria, Eric, Fang, Filip, Honghong, Hui, Jacque, Jingshuang,

Jon, Jumay, Lee Laoshi, Leon, Linda, Lucian, Mahesh, Nikos, Selcuk, Sho, Tom, Tudor, Vidhya, Yisong, Yunsong. Finally, Becky and Steph and the other administrative staff have been great, making sure that I filled out all my paperwork on time and generally making everything work smoothly the way it is supposed to. Thank you to everyone, and I'm sorry if I have omitted anyone's name.

Finally, this work was funded in part by NSF Career Award IIS-0237381, NSF Award OISE-0611783, NSF Grant IIS-0812091, and the KD-D grant.



## TABLE OF CONTENTS

Biographical Sketch . . . . .	iii
Dedication . . . . .	iv
Acknowledgements . . . . .	v
Table of Contents . . . . .	vii
List of Tables . . . . .	ix
List of Figures . . . . .	xi
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	1
1.2 Ideas in Text Documents . . . . .	2
1.3 Authorship as a Copy Process . . . . .	3
1.4 What Is Information Genealogy? . . . . .	4
1.4.1 Influence for Ideas . . . . .	5
1.4.2 Novelty for Ideas . . . . .	7
1.4.3 Original Ideas . . . . .	8
1.5 General Approach . . . . .	9
<b>2 Inter-Document Influence in Text</b>	<b>12</b>
2.1 Introduction . . . . .	12
2.2 Related Work: Measuring Influence . . . . .	14
2.2.1 Topic Detection and Tracking . . . . .	15
2.2.2 Real-World Influence on Documents . . . . .	16
2.2.3 Citation and Hyperlink Analysis . . . . .	16
2.2.4 Automatic Hypertext . . . . .	17
2.2.5 Language and Topic Models . . . . .	17
2.3 Methods . . . . .	18
2.3.1 Probabilistic Model and Motivation . . . . .	18
2.3.2 A Statistical Test for Detecting Influence . . . . .	21
2.3.3 Relating the LRT to Detecting Influence . . . . .	24
2.3.4 Computing the LRT . . . . .	24
2.4 Experiments . . . . .	26
2.4.1 Experiment Setup and Corpora . . . . .	26
2.4.2 Inferring Influence Graphs . . . . .	28
2.4.3 Identifying Influential Documents . . . . .	35
2.5 Discussion and Future Work . . . . .	40
2.6 Summary . . . . .	41
<b>3 Novel Ideas in Text Documents</b>	<b>42</b>
3.1 Introduction . . . . .	42
3.2 Related Work: Novelty Detection . . . . .	44
3.2.1 Novelty Detection in TDT . . . . .	44
3.2.2 TREC Novelty Track . . . . .	45

3.2.3	Other Novelty Tasks . . . . .	46
3.3	Task 1: Describing Novel Ideas in Their Own Words . . . . .	46
3.3.1	Method . . . . .	46
3.3.2	Experiments . . . . .	48
3.4	Task 2: Quantifying Each Document’s Novelty . . . . .	53
3.4.1	Method . . . . .	53
3.4.2	Experiments . . . . .	58
3.5	Discussion and Future Work . . . . .	72
3.6	Summary . . . . .	73
<b>4</b>	<b>Idea Origins in Text</b>	<b>74</b>
4.1	Introduction . . . . .	74
4.2	Related Work: Summarization and Novelty . . . . .	75
4.2.1	Document Summarization . . . . .	75
4.2.2	Novelty Detection . . . . .	76
4.2.3	Impact-Based Summaries . . . . .	76
4.2.4	Topic Modeling . . . . .	77
4.3	Methods . . . . .	78
4.3.1	Passage Impact Model . . . . .	78
4.3.2	Inference . . . . .	81
4.3.3	Implementation Details . . . . .	83
4.4	Experiments . . . . .	84
4.4.1	Analyzing the Model with Synthetic Data . . . . .	85
4.4.2	Predicting Quotations in Slashdot Discussions . . . . .	89
4.4.3	A User Study . . . . .	94
4.5	Discussion and Future Work . . . . .	96
4.6	Summary . . . . .	97
<b>5</b>	<b>Conclusions and Directions for Further Research</b>	<b>98</b>
5.1	Conclusions . . . . .	98
5.2	Future Work . . . . .	99
5.2.1	More Sophisticated Language Models . . . . .	100
5.2.2	Integration of Non-Textual Data . . . . .	100
5.2.3	Unified Model for Idea Structure Inference . . . . .	101
5.2.4	Evaluation and Collecting Data . . . . .	102
5.2.5	Generalization to Other Domains . . . . .	103
5.2.6	Scalability and Efficiency . . . . .	104

## LIST OF TABLES

2.1	Papers that are influenced by NIPS paper 1541, “Shrinking the Tube: a New Support Vector Regression Algorithm” written by B. Schoelkopf, P. Bartlett, A. Smola, and R. Williamson. The leftmost column shows the LRT statistic value. (Larger LRT statistic values represent greater influence.) . . . . .	28
2.2	G-MAP scores comparing the LRT against the similarity baseline. The similarity measure to select $\mathcal{P}$ is the TF cosine and to select/rank $C$ is either the TF cosine or the TFIDF cosine. Results are reported for $k = 100$ and $\sigma = 0.05$ . . . . .	32
2.3	G-MAP scores comparing the LRT for a range of $d^{(can)}$ influence mixing weights $\sigma$ against the similarity baseline. The similarity measure to select $C$ is either TF or TFIDF cosine. Results are reported on NIPS for $k = 100$ . . . . .	33
2.4	G-MAP scores comparing the LRT against the similarity baseline for two $k$ -NN approximation levels. The similarity measure for selecting $C$ is either TF or TFIDF cosine. Results are reported on NIPS and HEPH for $\sigma = .05$ . . . . .	34
2.5	How close is the approximation to the optimal? G-MAP scores are reported for $S = .05$ . . . . .	34
2.6	The most influential paper per year in NIPS, as measured by influence graph in-degree, with $k = 100$ , $\sigma = .05$ , and TFIDF cosine for $C$ . We exclude years with edge effects and the last 3 years, since they are not statistically significant. Comparison is against the within-NIPS citation counts, and Google-scholar citation counts (on May 26, 2009). . . . .	36
2.7	Rank metrics comparing the LRT against similarity on NIPS ( $k = 100$ ) and HEPH ( $k = 20$ ), using $\sigma = .05$ and TF or TFIDF cosine for $C$ . We ignore the first two and last two years because of edge effects. . . . .	37
3.1	Top 10 novel terms and highest-TFIDF (similarity baseline) terms for the yearly most-influential NIPS paper (Papers from Table 2.6). With $k_P = 100$ and $\pi_n^{(i)} = 0.05$ . . . . .	52
3.2	The most novel documents in NIPS according to the KL-Divergence score using Discount smoothing with $\delta = 0.01$ . . . . .	66
3.3	The most novel document per year of NIPS according to the KL-Divergence score using Discount smoothing with $\delta = 0.01$ . . . . .	67
3.4	The most novel documents in NIPS according to the KL-Divergence score, not including outlier documents, using Discount smoothing with $\delta = 0.01$ . . . . .	68
3.5	The most novel document per year of NIPS according to the KL-Divergence score, not including outlier documents, using Discount smoothing with $\delta = 0.01$ . . . . .	69

3.6	Basic statistics for the novelty score for years of NIPS, computed using Discount smoothing with $\delta = 0.01$ . . . . .	70
3.7	The KL-Divergence novelty score using Discount smoothing with $\delta = 0.01$ for the most influential paper per year of NIPS. The percentile is of the novelty score compared against all documents in that year. More details on how these papers are selected were presented in Table 2.6. . . . .	71
4.1	Percentage of misranked non-original passages. Passage length $L = 100$ , $\delta = 0.2$ , $\pi_n^{(i)} = 0.5$ , $\pi_i^{(l)} = 0.05$ , and $\pi_n^{(l)} = 0.6$ . 10 future documents $d^{(l)}$ were generated, and inference used the $k_F$ documents $d^{(l)}$ most (cosine) similar to $d^{(i)}$ . . . . .	87
4.2	Percentage of misranked non-original passages with $k_F = 2$ future documents. The data was generated with $\delta = 0.2$ , $\pi_n^{(i)} = 0.5$ , $\pi_i^{(l)} = 0.05$ , and $\pi_n^{(l)} = 0.6$ . . . . .	87
4.3	Percentage of misranked non-original passages. $k_F = 2$ future documents, passages length $L = 100$ words, $\pi_n^{(i)} = 0.5$ , $\pi_i^{(l)} = 0.05$ , and $\pi_n^{(l)} = 0.6$ . . . . .	88
4.4	Percentage of misranked non-original passages. $k_F = 2$ future documents, passage length $L = 100$ words, $\delta = 0.2$ , $\pi_n^{(i)} = 0.5$ , and $\pi_n^{(l)} = 0.6$ . . . . .	89
4.5	Prec@2 and Rec@10 are based on the predicted ranking of sentences by likelihood and TFIDF sum. Original sentences are the ones quoted word-for-word from the article. Results are for $\pi_n^{(i)} = 0.2$ and $\pi_n^{(l)} = 0.001$ . . . . .	92
4.6	Comparing the PIM with future documents, and PIM as a novelty detection method (without future documents). Results are for $\pi_n^{(i)} = 0.2$ and $\pi_n^{(l)} = 0.001$ . . . . .	92
4.7	Prec@2 and Rec@10 for various amounts of assumed novel content $\pi_n^{(i)}$ in $d^{(i)}$ . Sentences are marked as original if they appear word-for-word as in the linked article. Results are for $\pi_n^{(l)} = 0.001$ . . . . .	93
4.8	Prec@2 and Rec@10 for various mixing weights $\pi_n^{(l)}$ for the noise model in fitting future documents. Sentences are marked as original if they appear word-for-word as in the linked article. The results are reported for $\pi_n^{(i)} = 0.2$ . . . . .	94

## LIST OF FIGURES

2.1	ROC-Area comparing the LRT method against a cosine similarity baseline. The x-axis is $\pi_{can}^{(new)}$ . At a $\pi_{can}^{(new)}$ level, the ROC-Area measures the quality of influence prediction in documents with the specified $\pi_{can}^{(new)}$ as compared against documents with $\pi_{can}^{(new)} = 0$ . . . . .	27
2.2	Precision vs. Recall on NIPS. The three lines are (from top to bottom) the LRT method's precision at a recall level with TFIDF cosine used to select $C$ , the TFIDF distance $C$ similarity baseline, and the TF distance $C$ similarity baseline. . . . .	32
2.3	Using $\tau$ to compare the LRT against the similarity baseline, both with the $l$ parameter (left) and by thresholding the LRT statistic values (right). Results are for NIPS with TFIDF cosine $C$ and $k = 100$ . The TF plot looks similar, except that the baseline is smoother. . . . .	39
3.1	KL-divergence from the actual novel content distribution to the novel distribution learned according to the Inter-Document Influence Model (Model 2.3). The baseline shown is the KL-divergence from the actual novel language model to the MLE from the entire generated document. The x-axis is $\pi_n^{(i)}$ . At a $\pi_n^{(i)}$ level, the KL-divergence measures the amount of extra bits the inferred original content distribution (and baseline) need to encode the information in the true original content distribution. . . . .	49
3.2	KL-divergence from the actual novel content distribution to the learned novel distribution. The baseline is the KL-divergence from the actual novel language model to the MLE from the entire generated document. The generated document length was 100000 words. . . . .	50
3.3	ROC-Area analysis of the novelty score. The x-axis shows the amount of novel content $\pi_n^{(i)}$ in the generated documents. The graph shows ROC-Area of points ordered by the novelty score, for the various smoothing methods, with documents at this $\pi_n^{(i)}$ level being positive points and generated documents with $\pi_n^{(i)} = 0$ being negative points. The TFIDF baseline is based on taking the cosine distance from document $d^{(i)}$ to the single nearest document that precedes it in time. . . . .	58

3.4	This figure depicts the TFIDF Min distance baseline and why it may not work in the setting in which we generate documents. The groups of shaded boxes represent the MLEs from previous documents of similar topics, perhaps Neural Networks, Bayesian Methods, and Kernel Methods. The rounded box marked Mix in the middle represents the weighted mixture of the previous documents. The box marked Novel represents the novel language model. Here, as the novelty weight is increased in the generated documents, the generated documents will tend to lie along along the dotted line reaching from Mix to Novel. When Novel is similar to a previous document, the cosine distance between the TFIDF vectors of the generated document and the closest previous document may actually decrease with increasing novelty instead of increasing as would be intuitive. . . . .	59
3.5	ROC-Area analysis of the novelty score for Jelinek-Mercer smoothing with different $\lambda$ for controlling the amount of corpus smoothing. The baseline is TFIDF Max. . . . .	61
3.6	ROC-Area analysis of the novelty score for Dirichlet smoothing with different $\mu$ , compared against TFIDF Maximum Distance. . . . .	62
3.7	ROC-Area analysis of the novelty score for Discount smoothing with different $\delta$ , compared against TFIDF Maximum Distance. . . . .	62
3.8	ROC-Area analysis of the novelty score for Jelinek-Mercer smoothing with $\lambda = 0.01$ and 10, 30, and 100 previous documents. . . . .	64
3.9	ROC-Area analysis of the novelty score for Discount smoothing with $\delta = 0.01$ and 10, 30, and 100 previous documents. . . . .	64
4.1	The generative process for a corpus. Document $d^{(i)}$ is the current document, while $d^{(k)}$ precede $d^{(i)}$ in time and $d^{(l)}$ follow $d^{(i)}$ . The shaded boxes are original content $Z^{(i)}$ , while the rest of the documents form $\bar{Z}^{(i)}$ . The arrows depict the copy process. . . . .	80
4.2	Left: A post that quotes from article $d^{(i)}$ by the link “the way video games handle simulated emotions.” The label for the original content $z^{(i)}$ in $d^{(i)}$ is the quotation text. Right: Part of the discussion to be used as the future document $d^{(l)}$ . . . . .	90

# CHAPTER 1

## INTRODUCTION

### 1.1 Overview

In many domains, complete electronic records of documents now reach back for many years, even to several decades in some cases. Often, these document collections form by an archival process where, as documents are written, they accumulate and are stored for posterity. Many domains of document collections result from such an archival process, including research publications in journals and conferences, news articles, personal email, online discussion boards, blogs, websites in general, and so on.

People often interact with these document collections and thus may be interested in methods to help them better “use” the documents. For retrieving individual documents, search engines have already been very successful. Other methods such as topic modeling can provide a coarse overview of the topics in a document collection. While information retrieval and topic modeling methods have been widely applicable and useful, current methods for drilling deeper to understand the idea structure and development in a corpus as a whole could still be improved.

We provide methods for a set of tasks that seek to uncover the inter-document relationships by which ideas spread through a corpus over time. These methods focus on supplying a fine-grained picture of idea flows over time to help users grasp the document collection’s development as a whole. We will focus on several specific types of idea structures, including influential relationships between documents, novel ideas within documents, and original contributions of ideas from documents that can balance novelty and impact.

This thesis explores text-based language modeling approaches for these tasks. For wide applicability, the methods use only document text, instead of including hyper-link information which may be available for some corpora. We evaluate these methods experimentally on several corpora, including research publications from the Neural Information Processing Systems conference (NIPS Online, 2000). We have prepared a cleaned-up dataset with the text proceedings and compiled a citation graph between NIPS documents to conduct this evaluation.

## **1.2 Ideas in Text Documents**

We use the term “idea” broadly, encompassing things such as news events, contributions in research publications, or salient points in blog posts. Our hypothesis is that as humans write documents, their ideas are encoded in statistical properties of the text. Further, we assume (and will test experimentally) that analyzing these statistical properties of text can recover the structure of ideas in the corpus and make precise the interaction between documents and their ideas. In a sense, the assumption is that the statistical properties of text can serve as a “signature” of an idea. We hope that much as people’s signatures are intrinsic and basic to their identities, idea signatures are identifiable when they appear in documents, so that they can be used to recover the idea structure.

In analyzing ideas, the methods will use document text exclusively so that they apply widely to various document archives. Documents sometimes link to each other, e.g., citations in research publications, email replies or forwards, or blogs linking to other blogs. In many cases, however, the only data available is the text of the documents themselves. For example, news articles typically do not reference other news articles, blogs that discuss news events often do not reference specific news articles, email only



explicitly connects to itself, but not to the myriad events that cause people to send email, and even researchers cite existing papers for reasons other than giving credit for an existing idea (MacRoberts & MacRoberts, 1989; Baird & Oppenheim, 1994; Aya et al., 2005).

### **1.3 Authorship as a Copy Process**

As authors write documents, they seek to express their ideas through document text. By writing research publications, researchers propose novel and original content, while simultaneously responding to and building on existing ideas from the literature. Idea connections in research publications are often both implicit in the text and explicit in the form of citations. News articles have a similar authorship process in the sense that reporters write about new stories, while implicitly referring to past articles by summarizing the background context of the news story. Email and discussion boards, too, provide systems for proposing new ideas and discussing existing ideas.

These examples all depict diachronic document collections, which develop with a temporal dimension by being grown incrementally over time, instead of formed all at once. This temporal element allows such document archives to exhibit self-referential behavior where authors build on, respond to, and are influenced by existing ideas expressed in previous documents. Of course, documents may also introduce novel ideas. Generalizing the intuition from these examples to form an authorship model, documents seem to be written as authors introduce novel ideas or respond to existing ideas, or by some combination of these two. Here, responding to an existing idea is quite broad, including mentioning it, refuting it, developing it, or even connecting it with other ideas. We use the term “copy process” to refer to the model for how documents are written by

combining new and existing ideas.

In the copy process for ideas, on the one hand, authors express novel ideas – a new method or result in a research paper, a key point in a discussion on the web or over email, or a new topic for a blog – by writing text for that idea, which should then be distinguishable by its text signature. On the other hand, authors may reformulate existing ideas, essentially “copying” the ideas from the previous documents. This copying is on the idea level only, as in a newspaper article giving some background about an ongoing story along with giving the latest developments, or a research publication containing a related work section summarizing the contributions of previous work. The copying in this case is not word-for-word copying as in plagiarism detection (). Other ways of designing documents could also be expressed by the copy process. Extending and further developing an existing idea, e.g., by building a new method based on previous models in research publications, is a mixture where the author writes partially about the existing idea in explaining it and partially about the novel idea in the improvement of the method. Similarly, had the author written the document to refute an existing claim, the document would still contain a mixture of novel and “copied” ideas.

## **1.4 What Is Information Genealogy?**

To help people understand the structure of how ideas flow as they are addressed in other documents, we propose a set of questions that we call “information genealogy.” The word genealogy connotes exploring how ideas are picked up and explored throughout the collection of documents. In an ideal world, we could paint a picture showing the tracks that each idea takes, from its inception as a novel idea in some document to the last instance where this idea occurs. Text is quite noisy, and there are many documents,

so that we focus primarily on methods to help people grasp the most important ideas and how they developed. After all, not many people have the time to try to understand everything. In addition, people typically like to keep up-to-date with new and current events, or to see who first thought of an idea historically. Thus our methods typically, but not always, focus on the most important or earliest documents where ideas occur instead of trying to discover the documents that still contain ideas after they have passed their prime. This work therefore focuses on two specific properties of ideas: influence and novelty.

### **1.4.1 Influence for Ideas**

When writing, authors can refer to existing ideas from previous documents, thereby causing ideas to flow from earlier documents to the documents they are writing. In this scenario, we say that the earlier documents influenced the later documents, with the notion of influence corresponding to the copying of ideas. (Copying here means expressing or summarizing the existing idea, not plagiarism.) Another way of thinking about the copying or idea flow is that the text of the future document is influenced by the ideas of the earlier documents that contain those influential ideas. Research publications, especially, would seem to have explicit data for marking when ideas from one document influence another document. The citation is a mechanism by which authors can cite the sources from where they borrowed, built on, or otherwise responded to existing ideas. However, authors often cite for other reasons besides acknowledging a previous idea (MacRoberts & MacRoberts, 1989; Baird & Oppenheim, 1994; Aya et al., 2005). In addition, other collections may not have explicit link information, which argues for methods to detect these influence relationships by automatically analyzing text.

The task of detecting influence seems quite broad, so we focus on several specific questions in this work, which are the following: How can document text be used to detect when one document influences another document? How is it possible to recover the entire structure of the “influence graph” that shows which documents influence which other documents, somewhat analogous to a citation graph, but based on text rather than explicit citations? How is it possible to identify the most important or influential documents in the document collection?

We consider applications to set up the influence task. With the rapidly-increasing number of research articles published recently, one application is to automatically identify a small set, e.g., 10 or 20 research publications that have most influenced the content of the research field. We present such a method to identify the most influential research publications among the documents published at the Neural Information Processing Systems (NIPS) conference (NIPS Online, 2000). People who read this limited subset of articles could hopefully get the gist of the most important ideas and development of the research community. Reading recent influential publications could give the user a good sense of the latest trends and popular topics. As another example, for online discussion boards, a few particularly-insightful comments often stand out from the rest and spark much discussion. By starting with influential comments, the user could potentially save time by reading only the important comments instead of skimming the whole discussion. The influence method could also be used to visualize the development of ideas in the corpus. For example, one could make a graph where the documents are the vertices, and the (directed) edges are the strength of copying from between documents. This visualization would clearly depict the spread of the most important ideas. We will threshold the strength of influence to make an “influence graph” with edges between documents connected by a strong-enough influence relationship. Alternatively, graphing the relative popularity ideas over time gives a sense of how the collection evolves. Another

application for future work is based on the aggregation of the individual document and idea influence information, namely that by associating documents with their authors, the influence methods could be used to identify the authors that have contributed the most influential ideas. It would also be interesting to see whether authors consistently propose influential ideas or are known one or a few really great ideas that became highly influential.

### **1.4.2 Novelty for Ideas**

Apart from influence, we next address the task of identifying where novel ideas emerge. The methods we propose will detect what makes each document novel with respect to the documents that precede it in time. It will identify what is new and different about this document in terms of the novel ideas that it contains. In news articles, the existence of a novel idea may correspond to the occurrence of some news event. The first news article that breaks the story with the description of the facts is the document that proposes this new idea. For an ongoing story with developments, the novel idea in a document would be the latest development that no other article has yet reported on. In a collection of research publications, the novel ideas correspond quite naturally to the new research and results that the researchers have published. Intuitively, research publications should probably contain a large amount of novel content, as opposed to newspaper articles, which typically present one or a few new developments in an ongoing story, with only the occasional big story that is extremely novel.

The copy process, which will be made more precise later in the thesis, provides a formal framework for reasoning about and detecting novelty in text. With such methods, of the many specific questions that could be considered, we address the following: How

can a human-understandable overview of the identified novel ideas in each document be produced? How can the amount of novelty that each document has be quantified?

We consider real-life applications to inform how to answer these questions. With the popularization of word clouds and tagging in blogs, people seem to be growing more comfortable with browsing by keywords and sometimes just want a quick summary of the novel updates. We develop a method based on the models for influence to summarize the novel ideas in a document in terms of the most novel words from that document. For research publications, an interesting application would be to identify the documents that have the ideas with greatest novelty. Research publications by definition should be highly novel. The most novel documents could give the reader a starting point for understanding which publications are most pushing the field in new and hopefully promising directions. These most novel documents are the ones that propose radically new ideas, regardless of whether they eventually become popular or influence other documents. We develop a method based on information theory to quantify how much novelty each document contains.

### **1.4.3 Original Ideas**

As later experiments will show, novelty detection is not the right formulation for detecting important new ideas because the methods do not consider the impact of the novel ideas. We next combine novelty with influence to identify each document's original contribution. While novelty and influence each presents part of the picture of idea development, their combination may be much stronger for analyzing the idea structure. Looking at novel content only considers what is new or different about a document, without any regard for whether the idea is important. Looking at influence can identify

important ideas, but without any notion of novelty, influence alone cannot really identify the source for these important ideas. We combine novelty and influence and refer to it by the terms originality and original contributions. The original ideas in a document are defined to be those that are novel to that document and that eventually become important to the corpus.

Applications exist again for research publications. While the most novel publications are probably the ones that present the most unique methods, the most original publications are the ones that proposed novel ideas that then become popular. It would be extremely interesting to make a list of the most original research publications. As a further step, we will present a method to identify the passage within each document that has the most concentrated description of that document's original contribution that balances novelty and influence. For discussion boards or email discussions, typically with just one thread of discussion, originality detection can zero in on the particular comments that contain especially insightful new content that changed the course of the conversation or led to a dialogue. We apply the original content method to identifying the sentences from news articles that users select to begin discussions on the online discussion board Slashdot.

## **1.5 General Approach**

To address these tasks, we develop principled methods based on probabilistic language models of text. Using principled methods has the advantage that the methods are readily open to analysis, extension, and improvement. Previous approaches for the tasks that we will address have typically used various heuristics, with the most popular being Term Frequency (TF) combined with Inverse Document Frequencies (IDF) to form a

keyword-weighting scheme typically called TFIDF (). This heuristic in fact does quite well for applications in information retrieval (), text classification (), and document clustering (), among many others. The assumptions inherent in TFIDF, however, are not necessarily easy to express or analyze, although there have been analyses of TFIDF in the past (). By formalizing a precise generative model for text, the assumptions in our methods are obvious and easy to analyze. It also lends to extensibility, so that variants of these models and methods may potentially be used for other tasks or applications related to analyzing the idea structure.

We use methods that leverage only the text of the documents so that they apply to many domains of data. Many domains have no information about the idea structure besides what is expressed in the text of the documents themselves. Additionally, when there is idea transfer in heterogeneous document collections, e.g., blogs responding to news articles, or email responding to research publications, and so on, it is especially difficult to use some standard mechanism such as the citation to mark idea transfers.

Using the text only is a two-sided sword. While one advantage is that unsupervised methods that are based exclusively on text widely apply in many domains of documents, one disadvantage is that link information between documents also contains information that could be useful. For example, citation data for research publications could be used in addition to document text to find the most influential ideas. Citation data is not perfect for this task since people cite for various reasons, only one of which is to credit an existing idea (). Since text and citations are different types of data, however, the errors in these two types of data may cancel each other out to some extent, so that a method that uses all the data may do better than a method that uses only one type of data. While exploring combinations of text with other data is interesting, this thesis focuses on text-based methods and leaves such extensions for future work.



Evaluation of the methods is also not necessarily straightforward because of the lack of labeled data. Therefore, the manner of evaluation often depends on what type of data is available. In some cases, clever collection of particular real-world data allowed conducting the evaluation. At other times, we did user studies to see how actual users think these methods perform. Additionally, in most cases, we evaluate the methods on synthetically-generated data. Synthetic data allows close examination of the individual aspects of the models to see exactly how they might perform in different situations that may arise in real data. Finally, when it is insightful, we present qualitative evaluation of interesting cases or examples.

## CHAPTER 2

### INTER-DOCUMENT INFLUENCE IN TEXT

The first information genealogy task that we explore is understanding the connections between documents and their ideas, as manifested in the influence they have on each other. As the document collection grows over time, and authors draw on existing ideas in writing new documents, the text of these documents encodes these relationships. Specifically, when the author of some document uses an existing idea, there is an influence relationship from the earlier document that proposed the idea to the later document where the idea appears again. Starting from these inter-document influence relationships provides a text-based method for identifying the most influential documents in a corpus.

#### 2.1 Introduction

For self-referential document collections such as research publications, email, or news articles, we would like to answer the basic question: Did one document  $d$  influence another document  $d'$ ? This information can then be put together to answer more complicated interesting questions such as the following: What documents contain the most influential ideas? These documents are the most important ones in the collection, since they best represent the essence of the collection's ideas. Answering this fundamental question has many applications. On the web, methods such as Hubs and Authorities (Kleinberg, 1999) and PageRank (Page et al., 1998) have been used to find important documents. There is a whole research community that analyzes research publications by their citations to determine which have the most impact (Garfield, 1955; Garfield, 1972). Citations may not be the best data for measuring influence, however, because people cite documents for reasons besides acknowledging other important documents

that their work is related to. Other uses for this work could include judging whether citations were made to refer to influential ideas, suggesting citations for authors as they are writing, or inferring and visualizing idea flows relationships to help users browse the corpus.

Since much interesting text does not come with user-supplied hyperlink information, in contrast to bibliometric methods that are limited to collections with explicit citation structure, we investigate content-based methods requiring only the text and time stamps of the documents. Aggregating such information provides an algorithm that can use the text to infer the inter-document conduits through which ideas flow. Since ideas that spread more are by definition more influential, this temporal dependency structure between documents can then be used to make inferences about which documents are most influential.

The premise for this research is that ideas manifest themselves in statistical properties of a document (e.g., the distribution of words), and that these properties can act as a signature for an idea which can be traced through the database. Following this premise, we present a probabilistic model of influence between documents and design a content-based significance test to detect whether one document was influenced by an idea first presented in another document. The test takes the form of a Likelihood Ratio Test (LRT) and leads to a convex programming problem that can be solved efficiently. Our goal is to use this test for inferring an influence graph derived from the text of the documents alone.

Using corpora of scientific literature from the Neural Information Processing Systems Conference (NIPS) (NIPS Online, 2000) and the Physics ArXiv (Ginsparg, 1991), we show that it is indeed possible to infer meaningful influence graphs from the text of the documents. Evaluating against the explicit citation graphs for these corpora, we find

that the automatically-computed influence graphs are similar to the citation graphs. The ability to automatically generate an influence graph for a collection enables a range of applications, from browsing, to visualizing and mining the structure of the network. As a simple example, we demonstrate that the in-degree of the influence graph provides an interesting measure of document impact, similar to the in-degree of the citation graph. Furthermore, we show how that the Likelihood Ratio Test method based on the model for influence is more effective than methods based on document similarity.

## 2.2 Related Work: Measuring Influence

To begin, we investigate and operationalize the notion of influence between documents. Influence is an interesting relationship between documents in historically grown databases, since such corpora have grown through a self-referential process: documents are influenced by the content of prior documents, but also contribute new ideas which in turn influence later documents. Our goal is to uncover and mine how ideas introduced in some document spread through the corpus over time.

At first glance, one might think that similarity, as captured by information retrieval metrics like TFIDF cosine similarity (e.g., (Salton & Buckley, 1988)), provides the full picture of influence. However, this is not the case.

On the one hand, similarity can occur without influence. First, if a document  $d^{(1)}$  introduces an idea that is picked up in documents  $d^{(2)}$  and  $d^{(3)}$ , then  $d^{(2)}$  and  $d^{(3)}$  will likely be similar but do not necessarily influence each other. Second, two documents might concurrently propose the same idea. Again, neither document influences the other although the documents likely are similar.

On the other hand, influence can occur with very little similarity. In the scientific literature, for example, a large textbook might devote a section to an idea introduced in an earlier research paper. Clearly, the paper had influence on the textbook. However, the overall similarity between the book and the paper is small, since the book covers many other ideas as well.

As we will briefly review in the following, most prior work on analyzing temporal corpora has focused on identifying relatedness between documents, not influence. We will then develop a probabilistic model and a statistical test for detecting influence, and show that it captures influence better than similarity and provides a more complete understanding and model of influence.

### **2.2.1 Topic Detection and Tracking**

Topic Detection and Tracking (TDT) (Allan et al., 1998a; Allan et al., 1998b) has the goal of grouping documents by topic. Unlike influence, which is a directed relationship, TDT aims to group documents into equivalence classes. While TDT approaches have relied heavily on finding similarity measures that capture closeness in topic, this approach is not necessarily detecting influence, as we have argued above. Methods that model influence not only can detect and track topics and ideas, but also can provide reference points for *why* a document collection developed as it did. Another minor difference is that the TDT studies were performed in an online setting, while we assume access to the full corpus at any time.

Similar work on detecting and visualizing topic development includes visualization methods such as Temporal Cluster Histograms (Shaparenko et al., 2005) and ThemeRiver (Havre et al., 2002), EM-based corpus evolution detection (Mei & Zhai, 2005),

temporal clustering methods (Blei & Lafferty, 2005; Wang & McCallum, 2006), continuous time clustering models (Wang & McCallum, 2006), Thread Decomposition (Guha et al., 2005), Independent Component Analysis (Kolenda et al., 2001), topic-intensity tracking (Krause et al., 2006), and Topical Precedence (Mann et al., 2006).

## **2.2.2 Real-World Influence on Documents**

Research on Burst Detection (Kleinberg, 2002) and TimeMines (Swan & Jensen, 2000) aims to identify hidden causes based on changes in the word distribution over time. However, their notion of influence is different from ours. These approaches determine influence from real-world events on topics (e.g., events influencing US State of the Union Addresses). Instead, we model the influence of documents on each other.

## **2.2.3 Citation and Hyperlink Analysis**

In bibliometrics, a document’s influence is measured through properties of the citation graph (Osareh, 1996; Page et al., 1998; Kleinberg, 1999; Garfield, 2003). Our work differs from citation analysis because our method is based on document content, not on citations. We assume that influence is inherently reflected in the statistical properties of documents. In particular, we conjecture that when one document influences another, the influenced document shows traces of the word distribution of the earlier document<sup>1</sup>. Besides bibliometrics’ consideration of citation analysis on research papers, other methods work on general hyperlink structure. One of the most well-known such methods is PageRank (Page et al., 1998), which uses hyperlink structure to find influential Web

---

<sup>1</sup>Note that our goal is not plagiarism detection, where authors would try to disguise their choice of words.

pages.

#### **2.2.4 Automatic Hypertext**

There is related work on automatically adding hyperlinks in information retrieval and related fields. Most prominently, Link Detection was a key task in the TDT evaluations (Allan et al., 1998a). Several proposals and methods exist for introducing hyperlinks between similar documents or passages of documents (Furuta et al., 1989; Coombs, 1990; Salton & Buckley, 1991; Lelu, 1991; Agosti & Crestani, 1993; Allan, 1995; Agosti et al., 1997; Kurland & Lee, 2004; Kurland & Lee, 2006). Furthermore, the problem of detecting different types of links was considered in (Allan, 1997) and in (Aya et al., 2005). Good surveys are given in (Wilkinson & Smeaton, 1999) and the 1997 special issue of *Information Processing and Management* (Agosti & Allan, 1997). The work we propose is different in several respects. First, our goal is to detect influence between documents, not just their “relatedness.” This will allow a causal interpretation of the resulting citation graph. Second, we take a statistical testing approach to the problem of identifying influence links, which can be seen as synonymous to citations. This will give a formal semantic to the predictions of the methods, give theoretical guidance on how to apply the methods, and expose underlying assumptions.

#### **2.2.5 Language and Topic Models**

We take a probabilistic language modeling approach in the development of our methods. While we rely on a rather basic language model for the sake of simplicity, more detailed language models exist and can possibly be employed as well. Previous work

by Steyvers et al. (Steyvers et al., 2004) looks at how document text can be generated by a two-step model of generating topics probabilistically from authors, and then words probabilistically from topics. There has also been language modeling work done in the natural language processing and machine learning (Manning & Schuetze, 1999; Hofmann, 1999; Blei et al., 2003b), speech recognition (Jelinek, 1998), and information retrieval communities (Zhai, 2002; Kurland & Lee, 2004; Kurland & Lee, 2006).

## 2.3 Methods

In constructing an influence graph for a database of documents, the core problem is to determine when and where ideas flow from one document to another document. In the following, we propose a probabilistic model of influence in a language-modeling framework, and develop a Likelihood Ratio Test (LRT) (Casella & Berger, 2002) for detecting whether one document has significantly influenced another document.

### 2.3.1 Probabilistic Model and Motivation

To make the method widely applicable, we have only two basic requirements for our corpus of documents — first, the documents contain text and, second, the documents have time stamps. Formally, the corpus  $\mathcal{D}$  is a collection of  $n$  documents  $\{D^{(1)} \dots D^{(n)}\}$ , where each document  $D^{(i)} \in \mathcal{D}$  has an associated time stamp  $time(D^{(i)})$ . The number of unique terms (words) in the corpus is denoted by  $m$ .

We assume that each document  $D^{(i)}$  is a vector-valued random variable of  $n_i$  words, i.e.,  $D^{(i)} = (W_1^{(i)} \dots W_{n_i}^{(i)})$ . This notation describes a document as a sequence of word random variables  $W_j^{(i)}$ . A particular observed document is denoted as  $d^{(i)} = (w_1^{(i)} \dots w_{n_i}^{(i)})$ .



In the following, we assume that each document  $D^{(i)} \in \mathcal{D}$  was generated by drawing these words  $P(D^{(i)} = d^{(i)} | \theta^{(i)})$  from a unigram language model with parameters  $\theta^{(i)}$  specific to document  $d^{(i)}$ .

**Model 2.1** (DOCUMENT LANGUAGE MODEL)

*A document  $D^{(i)} \in \mathcal{D}$  is assumed to be generated by independently drawing  $n_i$  words from a document-specific distribution with individual word probabilities parameterized by  $\theta^{(i)}$ , i.e., that*

$$\begin{aligned} P(D^{(i)} = d^{(i)} | \theta^{(i)}) &= P(D^{(i)} = (w_1^{(i)} \cdots w_{n_i}^{(i)}) | \theta^{(i)}) \\ &= \prod_{j=1}^{n_i} P(W_j^{(i)} = w_j^{(i)} | \theta^{(i)}) \\ &= \prod_{j=1}^{n_i} \theta_{w_j^{(i)}}^{(i)} \end{aligned}$$

We chose this basic language model for mathematical and computational convenience. However, our approach can be extended to more complex language models as well (e.g., n-gram models).

Since we wish to detect the flow of ideas and influence between documents, we also need a model for inter-document relationships. We formalize this as a question of how the language model  $\theta^{(i)}$  of a new document  $D^{(i)}$  depends on the documents  $d^{(k)}$  that precede  $D^{(i)}$  in time. The set of previous documents  $d^{(k)}$  is indexed by the set  $\mathcal{P} = \{k : \text{time}(D^{(k)}) < \text{time}(D^{(i)})\}$ . Since the actual document language models  $\theta^{(k)}$  are unknown for these documents  $d^{(k)}$ , we use the maximum likelihood estimator  $\hat{\theta}^{(k)}$  based on the word distribution of  $d^{(k)}$ . This mixture is a linear combination controlled by document-specific mixing weights  $\pi^{(i)}$ . In short, we assume that the language model  $\theta^{(i)}$  of a new document  $D^{(i)}$  can be (approximately) expressed as a mixture distribution over the language models  $\hat{\theta}^{(k)}$  of previous documents  $d^{(k)}$  with mixing weights  $\pi^{(i)}$ . We formalize this assumption in the following model:

**Model 2.2** (INTER-DOCUMENT INFLUENCE MODEL)

A new document  $D^{(i)}$  is generated by a mixture distribution of the already existing documents  $D^{(k)}$  with  $k \in \mathcal{P}$  for previous document indices  $\mathcal{P} = \{k : \text{time}(d^{(k)}) < \text{time}(D^{(i)})\}$ , in particular

$$P(D^{(i)} = d^{(i)} | \pi^{(i)}) = \prod_{j=1}^{n_i} \sum_{k \in \mathcal{P}} \pi_k^{(i)} \hat{\theta}_{w_j^{(i)}}^{(k)} \quad (2.1)$$

with mixing weights  $\pi^{(i)}$  satisfying  $0 \leq \pi_k^{(i)}$  and  $\sum_k \pi_k^{(i)} = 1$ .

In this dependency model, a new document is composed of parts generated by the word distributions of old documents, where the mixing coefficient  $\pi_k^{(i)}$  indicates the fraction of  $D^{(i)}$  that is generated from the old document  $d^{(k)}$ . Clearly, there is direct influence from an existing previous document  $d^{(k)}$  on  $D^{(i)}$ , if the respective mixing coefficient is non-zero. The resulting language model for  $D^{(i)}$  is a unigram model, so that  $P(D^{(i)} = d^{(i)} | \pi^{(i)}) = P(D^{(i)} = d^{(i)} | \theta^{(i)})$  with

$$\theta^{(i)} = \sum_{k \in \mathcal{P}} \pi_k^{(i)} \hat{\theta}^{(k)}. \quad (2.2)$$

Actual documents typically contain some novel content that does not come from previous documents. To account for document novelty in our model, we include a novel language model  $\bar{\theta}^{(i)}$  with weight  $\pi_n^{(i)}$  in the mixture for  $D^{(i)}$ . This distribution models words that are novel to  $D^{(i)}$  and that cannot be explained by previous documents. (In practice, we will assume that  $\pi_n^{(i)}$  is fixed, but that we have no knowledge of  $\bar{\theta}^{(i)}$ .)

**Model 2.3** (INTER-DOCUMENT INFLUENCE MODEL WITH NOVEL CONTENT)

A new document  $D^{(i)}$  is generated by a mixture distribution of the already existing documents  $D^{(k)}$  with  $k \in \mathcal{P}$  for previous documents indices  $\mathcal{P} = \{k : \text{time}(d^{(k)}) < \text{time}(D^{(i)})\}$ , and a document-specific novel component  $\bar{\theta}^{(i)}$  with weight  $\pi_n^{(i)}$ , in particular

$$P(D^{(i)} = d^{(i)} | \pi^{(i)}) = \prod_{j=1}^{n_i} \left( \pi_n^{(i)} \bar{\theta}_{w_j^{(i)}}^{(i)} + \sum_{k \in \mathcal{P}} \pi_k^{(i)} \hat{\theta}_{w_j^{(i)}}^{(k)} \right) \quad (2.3)$$

with mixing weights  $\pi^{(i)}$  s.t.  $0 \leq \pi_k^{(i)}, \pi_n^{(i)}$  and  $\pi_n^{(i)} + \sum_k \pi_k^{(i)} = 1$ .

In the case when the documents have no novel content, setting  $\pi_n^{(i)} = 0$  in the Inter-Document Influence Model with Novel Content results in Model 2.2. Vice versa, Model 2.2 also subsumes Model 2.3 by simply introducing an artificial single-word document for each term in the corpus and constraining their mixture weights to sum to  $\pi_n^{(i)}$ . We will therefore focus our further derivations on Model 2.2 for the sake of simplicity.

We will now show how this probabilistic setup can be used in a significance test for detecting whether a particular mixing weight  $\pi_k^{(i)}$  is non-zero in a given document collection.

### 2.3.2 A Statistical Test for Detecting Influence

How can one decide whether a candidate influential document  $d^{(can)}$  had a significant influence on  $d^{(new)}$  given the other documents in the collection? First,  $d^{(can)}$  can only have had an influence on  $d^{(new)}$  if it had been published before  $d^{(new)}$  (i.e.,  $time(d^{(can)}) < time(d^{(new)})$ ). Note that this is already encoded in the Inter-Document Influence Models defined above. Second, influence should be attributed to the first publication that introduced an idea through an novel section or portion, not to other documents that later copied an idea. To illustrate this in the context of research papers, this means that influence should be credited to the earlier article, not a tutorial that reproduced the novel idea.

Under these conditions, the decision of whether document  $d^{(new)}$  shows significant influence from  $d^{(can)}$  can be phrased as a Likelihood Ratio Test (Casella & Berger, 2002). In general, a Likelihood Ratio Test decides between two families of densities described

by sets of parameters  $\Pi$  and  $\Pi_0$  that are nested, i.e.,  $\Pi_0 \subset \Pi$ . Applied to our case,  $\Pi$  will be all mixture models of  $D^{(new)}$  as in Eq. (2.1) with parameters  $\pi^{(new)}$  for all documents  $\mathcal{P}$  published prior to  $t_0 = \text{time}(d^{(can)})$  (and therefore prior to  $d^{(new)}$ ), as well as a parameter  $\pi_{can}^{(new)}$  for  $d^{(can)}$ .

$$\Pi = \left\{ \pi^{(new)} : \pi_{can}^{(new)} + \sum_{k \in \mathcal{P}} \pi_k^{(new)} = 1 \quad \wedge \quad \pi_k^{(new)} \geq 0 \quad \wedge \quad \pi_{can}^{(new)} \geq 0 \right\}$$

The subset  $\Pi_0$  of the mixture models in  $\Pi$  will be the models where  $d^{(can)}$  has zero mixture weight (i.e.,  $\pi_{can}^{(new)} = 0$ ).

$$\Pi_0 = \left\{ \pi^{(new)} : \pi_{can}^{(new)} + \sum_{k \in \mathcal{P}} \pi_k^{(new)} = 1 \quad \wedge \quad \pi_k^{(new)} \geq 0 \quad \wedge \quad \pi_{can}^{(new)} = 0 \right\}$$

Note that the set of prior documents  $\mathcal{P} = \{k : \text{time}(d^{(k)}) < \text{time}(d^{(can)})\}$  serves as a “background model” of what was already known when  $d^{(can)}$  was published. Against this background, we can then measure how much the new ideas in document  $d^{(can)}$  influenced  $d^{(new)}$ .

The null hypothesis of the Likelihood Ratio test is that the data comes from a model in  $\Pi_0$  (i.e., document  $d^{(new)}$  was not influenced by  $d^{(can)}$  given the documents published before  $d^{(can)}$ ). To reject this null hypothesis, a likelihood ratio test considers the following test statistic

$$\Lambda_{d^{(can)}}(d^{(new)}) = \frac{\sup_{\pi \in \Pi_0} \{P(D^{(new)} = d^{(new)} \mid \pi)\}}{\sup_{\pi' \in \Pi} \{P(D^{(new)} = d^{(new)} \mid \pi')\}}$$

Note that  $P(D^{(new)} = d^{(new)} \mid \pi)$  is convex over  $\Pi$  and  $\Pi_0$ , so that the suprema can be computed efficiently. We will elaborate on the computational aspects below. Intuitively, the value of  $\Lambda_{d^{(can)}}(d^{(new)})$  measures whether using  $d^{(can)}$  in the mixture model better explains the content of  $d^{(new)}$  than just using previously published documents. More formally,  $\Lambda_{d^{(can)}}(d^{(new)})$  compares the likelihood  $\sup_{\pi' \in \Pi} \{P(D^{(new)} = d^{(new)} \mid \pi')\}$  of the best mixture model containing  $d^{(can)}$  with the likelihood  $\sup_{\pi \in \Pi_0} \{P(D^{(new)} = d^{(new)} \mid \pi)\}$  of the best

mixture model that does not use  $d^{(can)}$  (i.e.,  $\pi_{can}^{(new)} = 0$ ). The test then decides whether there is significant evidence that a non-empty part of  $d^{(new)}$  was generated from  $d^{(can)}$ , in comparison to using a mixture only over the other language models.

If the null hypothesis is true, then the distribution of the LRT statistic  $-2 \log(\Lambda_{d^{(can)}}(d^{(new)}))$  is asymptotically (in the document length under the unigram model)  $\chi^2$  with one degree of freedom.

$$-2 \log(\Lambda_{d^{(can)}}(d^{(new)})) \sim \chi_1^2$$

The null hypothesis  $H_0$  should be rejected, if

$$-2 \log(\Lambda_{d^{(can)}}(d^{(new)})) > c$$

for some  $c$  selected dependent on the desired significance level. For a significance level of 95%,  $c$  should be 3.84. This captures the intuition that we can reject the null hypothesis and conclude that  $d^{(can)}$  had a significant influence on  $d^{(new)}$ , if the best model that does not use  $d^{(can)}$  has a much worse likelihood than the best model that considers  $d^{(can)}$ . Specifically, if  $-2 \log(\Lambda_{d^{(can)}}(d^{(new)}))$  is large, then  $d^{(can)}$  significantly influenced  $d^{(new)}$  given all other documents published at that time.

To estimate the language models  $\hat{\theta}^{(k)}$  of the previous documents  $d^{(k)}$  used in the mixture model of  $d^{(new)}$ , we use the maximum-likelihood estimator. We denote with  $tf^{(k)}$  the term frequency (TF) vector of document  $d^{(k)}$ , where each entry  $tf_w^{(k)}$  is the number of times that word  $w$  appears in the document  $d^{(k)}$ . The estimator is

$$\hat{\theta}_w^{(k)} = \frac{tf_w^{(k)}}{n_k},$$

which is simply the fraction of times the particular word occurs in the observed document  $d^{(k)}$ . Using a more advanced estimator instead is straightforward, but we will not discuss this for the sake of simplicity.

### 2.3.3 Relating the LRT to Detecting Influence

What does it mean for the LRT to significantly reject the null hypothesis? A good intuition is to think of this method in the context of trying to explain the ideas and content found in  $d^{(new)}$ . There are two choices. First, explain  $d^{(new)}$  using only other documents preceding  $d^{(can)}$  as well as some novel component. Second, explain  $d^{(new)}$  with these plus an additional  $d^{(can)}$ . If the first case already provides a wonderful model for  $d^{(new)}$ , then adding  $d^{(can)}$  will not explain  $d^{(new)}$  any more accurately. Thus,  $d^{(can)}$  really does not contribute to  $d^{(new)}$ . On the other hand, if  $d^{(can)}$  introduced some new ideas and terminology that then flowed to  $d^{(new)}$ , using  $d^{(can)}$  will provide a better explanation than only using  $\mathcal{P}$ . Consequently, the likelihood of  $d^{(new)}$  using  $d^{(can)}$  will be significantly higher than without it, and we can reject the null hypothesis. To summarize, rejecting the null hypothesis means that  $d^{(can)}$  significantly exerted influence on  $d^{(new)}$ .

### 2.3.4 Computing the LRT

Computing the value of  $\Lambda_{d^{(can)}}(d^{(new)})$  requires solving two optimization problems.

$$L_0 = \sup_{\pi \in \Pi_0} \{P(D^{(new)} = d^{(new)} \mid \pi)\} \text{ and} \quad (2.4)$$

$$L = \sup_{\pi \in \Pi} \{P(D^{(new)} = d^{(new)} \mid \pi)\}. \quad (2.5)$$

Given our model, these problems can be solved efficiently. Note that we can write the log-likelihood  $L(\pi \mid d^{(new)}, \mathcal{S})$  of the document  $d^{(new)}$  w.r.t. a fixed  $\pi$  as

$$\begin{aligned} \log L(\pi \mid d^{(new)}, \mathcal{S}) &= \log P(d^{(new)} \mid \pi, \mathcal{S}) \\ &= \sum_{j=1}^{n_{new}} \log \left( \sum_{k \in \mathcal{S}} \pi_k^{(new)} \hat{\theta}_{w_j^{(k)}}^{(k)} \right) \\ &= \sum_{w \in V} t f_w^{(new)} \log \left( \sum_{k \in \mathcal{S}} \pi_k^{(new)} \hat{\theta}_w^{(k)} \right). \end{aligned} \quad (2.6)$$

With  $\mathcal{S}$  we denote the set of documents considered in the model. This gives  $\mathcal{S} = \mathcal{P} \cup \{can\}$  for  $\Pi$  and  $\mathcal{S} = \mathcal{P}$  for  $\Pi_0$ . In this notation, each of the optimization problems in Eq. (2.4) and (2.5) takes the form

$$\begin{aligned} & \max_{\pi \in \mathbb{R}^{|\mathcal{S}|}} && \log L(\pi \mid d^{(new)}) \\ \text{subject to} &&& \sum_{k \in \mathcal{S}} \pi_k^{(new)} = 1 \\ &&& \forall k \in \mathcal{S} : \pi_k^{(new)} \geq 0. \end{aligned}$$

For Model 2.3, the mixture in the likelihood contains the additional term  $\pi_n^{(new)} \bar{\theta}_{w_j}^{(new)}$ . There is also an additional linear constraint is introduced to limit the amount of novel content  $\pi_n^{(new)}$  to not be more than a user-specified parameter  $\sigma$ . This constraint is necessary, since otherwise the  $\bar{\theta}^{(new)}$  mixture component could always perfectly explain  $d^{(new)}$ .

It is easy to see that these optimization problems are convex, which means that they have no local optima and that there are efficient methods for computing the solution. We currently use the separable convex implementation for the general-purpose solver Mosek (MOSEK, 2008) to solve the optimization problems. However, more specialized code is likely to be substantially more efficient.

While solving each optimization problem is efficient, analyzing a collection requires a quadratic number of LRTs, each with on the order of  $n$  documents in the background model. In particular, for each document  $d^{(new)}$ , we need to test all prior documents

$$C = \{d^{(k)} : time(d^{(k)}) < time(d^{(new)})\} \quad (2.7)$$

in the collection, since all of these are candidates for having influenced  $d^{(new)}$ . For each document  $d^{(can)}$  in the candidate candidate set  $C$  of  $d^{(new)}$ , we then have a background model

$$\mathcal{P}_{d^{(can)}} = \{d^{(k)} : time(d^{(k)}) < time(d^{(can)})\}. \quad (2.8)$$

Computing all tests exhaustively for a large corpus can be expensive. We therefore use the following approximations.

Both approximations are based on the insight that some similarity is typically present in documents joined by an influence relationship. The potentially influential document  $d^{(can)}$  should have some similarity with  $d^{(new)}$ . Therefore, we first approximate the candidate set to contain the  $k_C$  nearest neighbors of  $d^{(new)}$  from  $C$ . We use cosine distance between TF and TFIDF vectors for document similarity. Second, an analogous argument applies to the background models  $\mathcal{P}_{d^{(can)}}$ . We therefore approximate the background model, using only the  $k_P$  most similar documents from  $\mathcal{P}$ . Since selecting  $\mathcal{P}$  combines document vectors by addition, we use cosine distance between document TF vectors to select  $\mathcal{P}$ . In the experiments we set  $k_C = k_P$  and refer to this parameter as  $k$ . We will empirically evaluate the effect of these approximations depending on  $k$ .

## 2.4 Experiments

We wish to measure how well these models' assumptions match real data. First, how does an influence graph inferred by the LRT method compare against a citation graph? Second, can the influence graph identify top influential papers?

### 2.4.1 Experiment Setup and Corpora

The concept of influence and idea flow between documents corresponds well with the notion of a citation. Consequently, we focus on research papers to provide a quantitative evaluation of the LRT method by comparing with citations.



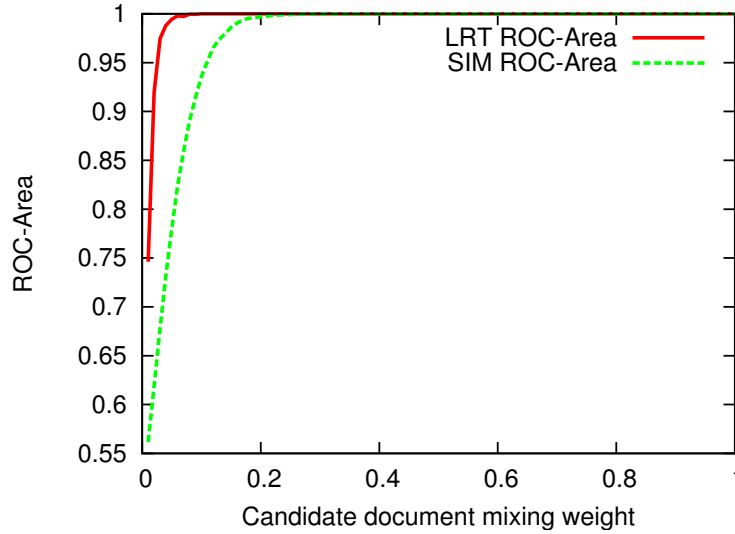


Figure 2.1: ROC-Area comparing the LRT method against a cosine similarity baseline. The x-axis is  $\pi_{can}^{(new)}$ . At a  $\pi_{can}^{(new)}$  level, the ROC-Area measures the quality of influence prediction in documents with the specified  $\pi_{can}^{(new)}$  as compared against documents with  $\pi_{can}^{(new)} = 0$ .

The first corpus is the full-text proceedings of the Neural Information Processing Systems (NIPS) conference (NIPS Online, 2000) from 1987-2000, with a time stamp of the publication year. NIPS has 1955 documents, with 74731 terms (features). We manually constructed the graph of 1512 intra-corpus citations, but only compare to citations of previous documents in time. We ignore citations of first-year documents since the LRT requires a background model.

The second corpus is the theoretical high-energy physics (HEPTH) section of the Physics ArXiv (Ginsparg, 1991) from Aug. 1991 to Apr. 2006. We aggregate the full-text papers by year. HEPTH has 39008 documents, 229194 terms, and 557582 citations. SLAC-SPIRES compiled these citations.

Table 2.1: Papers that are influenced by NIPS paper 1541, “Shrinking the Tube: a New Support Vector Regression Algorithm” written by B. Schoelkopf, P. Bartlett, A. Smola, and R. Williamson. The leftmost column shows the LRT statistic value. (Larger LRT statistic values represent greater influence.)

$-2 \log(\Lambda_{d(1541)}(d^{(new)}))$	Cite?	Title and Author(s) of $d'$
321.2455	no	“Support Vector Method for Novelty Detection” by B. Schoelkopf, Robert C. Williamson, Alex Smola, John Shawe-Taylor, John C. Platt
221.8297	yes	“An Improved Decomposition Algorithm for Regression Support Vector Machines” by Pavel Laskov
219.8769	yes	“ $\nu$ -arc: Ensemble Learning in the Presence of Outliers, Gunnar Raetsch” by B. Scholkopf, Alex Smola, Kenneth D. Miller, Takashi Onoda, Steve Mims
184.5493	no	“Fast Training of Support Vector Classifiers” by Fernando Perez-Cruz, Pedro Alarcon-Diana, Angel Navia-Vazquez, Antonio Artes-Rodriguez
168.8972	yes	“Uniqueness of the SVM Solution” by Christopher J. C. Burges, David J. Crisp

## 2.4.2 Inferring Influence Graphs

This set of experiments analyzes how well the LRT recovers the influence graph. After an illustrative example, we explore the LRT’s sensitivity on synthetic data under controlled experiment conditions, and then evaluate on two real-world datasets.

## Qualitative Evaluation

We first discuss a simple example to illustrate the LRT method’s behavior and how it compares to citations. Table 2.1 shows those documents that NIPS document 1541 (Schoelkopf et al. on “Shrinking the Tube: a New Support Vector Regression Algorithm”) most significantly influenced according to the LRT statistic. Three of the top five papers actually cite document 1541 (or a document with equivalent content from another venue). Furthermore, the top document could arguably have cited 1541 as well, since it relies on the  $\nu$ -parameterization of SVMs that document 1541 introduced to NIPS. In fact, all papers (except “Fast Training of Support Vector Classifiers”) consider this new parameterization. Note that the paper “ $\nu$ -arc: Ensemble Learning in the Presence of Outliers” is not about SVMs, but uses the  $\nu$ -parameterization in the context of boosting.

The LRT appears to accurately focus on the paper’s novel contribution, the  $\nu$ -parameterization. General SVM papers do not score highly, since they are already modeled by earlier papers, e.g., paper 1217 “Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing” of V. Vapnik et al., which was one of the first SVM papers in NIPS. When considering influencers of “A Support Vector Method for Clustering” by A. Ben-Hur et al. (using the conventional parameterization), the method correctly recognizes that paper 1541’s influence is low ( $-2 \log(\Lambda_{d^{(1541)}}(d^{(new)})) = 67.0$ ) even though the documents are similar. Paper 1217 already “explains” the SVM content ( $-2 \log(\Lambda_{d^{(1217)}}(d^{(new)})) = 535.0$ ).

## Quantitative Evaluation on Synthetic Data

Beyond this qualitative example, how accurately can the LRT discover influence? How much must  $d^{(new)}$  copy from  $d^{(can)}$  before the LRT can detect it?

To explore these questions, we constructed synthetic documents  $d^{(new)}$  from the NIPS corpus as follows. A candidate document  $d^{(can)}$  and a set  $\mathcal{P}$  of  $k = 100$  previous documents are chosen at random from the NIPS corpus so that the documents in  $\mathcal{P}$  precede  $d^{(can)}$  in time. Then, 101 artificial new documents are generated according to Eq. 2.1, where each new document has been influenced by  $d^{(can)}$  at the fractional levels of  $\pi_{can}^{(new)} \in \{0.00, 0.01, 0.02, \dots, 1.00\}$ . The remaining mixing weights  $\pi_k^{(new)}$  are selected by generating random numbers uniformly on the interval  $[0, 1]$ , and then normalizing them so that they sum to  $1 - \pi_{can}^{(new)}$ . The LRTs are run on each new document. Additionally, TF document vector cosine similarity is measured between  $d^{(can)}$  and each  $d^{(new)}$ . The entire process is repeated for 1000 random selections of  $\mathcal{P}$  and  $d^{(can)}$ .

We computed ROC-Area in the following manner. First, we select a particular  $\pi_{can}^{(new)} \in \{0.01 \dots 1.00\}$ . The generated documents at the  $\pi_{can}^{(new)}$  level are marked as positive examples. The negative examples are documents with  $\pi_{can}^{(new)} = 0$ . Finally, a ranking, either LRT statistic scores or cosine distance similarity, is used to compute ROC-Area.

Figure 2.1 shows that even if only a small portion (i.e., a few percent) of  $d^{(new)}$  is drawn from  $d^{(can)}$ , the LRT accurately detects the influence. The similarity baseline needs a much larger signal. This example illustrates that similarity and influence are in fact different, and that the well-founded statistical approach can be more accurate and sensitive than an ad-hoc heuristic.

## Quantitative Evaluation on Real Data

Moving to real data, we use the LRTs to discover the influence graph for NIPS and HEPATH. For each document  $d^{(new)}$ , we first compute a set of candidate documents  $C$  based on similarity. The elements of  $C$  are then ranked according to the LRT statistic (i.e., whether  $d^{(can)}$  was significant in explaining  $d^{(new)}$ ). The higher  $d^{(can)}$  is ranked, the more likely that it influenced  $d^{(new)}$ , and we can derive the influence graph by thresholding (discussed below).

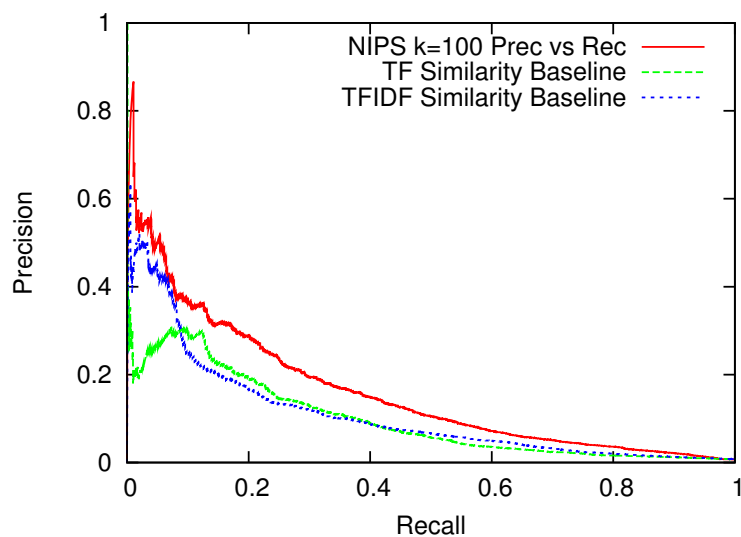
We evaluate the influence graph by a graph-based mean-average-precision (G-MAP) metric. For a document  $d$ , average the precision of the ranked predicted list of influencers at the positions corresponding to documents that  $d$  actually cites. Citations not in the list are averaged as 0, i.e., ranked at infinity. (As an information retrieval analogy, the influence list is the search result page, with citations being relevant results.) G-MAP is the mean of the per-document average precision scores. We exclude documents from the first two years due to edge effects (the LRT cannot predict citations for the first years since  $C$  or  $\mathcal{P}$  are empty).

We compare G-MAP for the LRT method against G-MAP of a similarity-based heuristic, which serves as a baseline. This baseline method ranks the elements of  $C$  not by LRT score, but by similarity. We explored several similarity measures. The best similarity measures in our experiments are TF cosine and TFIDF cosine. We report their performance.

Note that citations are not necessarily a perfect gold standard for influence, since they reflect idiosyncracies of how scientific communities cite prior work. For example, in Table 2.1 authors sometimes cited a journal paper or book instead of the NIPS paper. Therefore, a G-MAP of 1 is not achievable.

Table 2.2: G-MAP scores comparing the LRT against the similarity baseline. The similarity measure to select  $\mathcal{P}$  is the TF cosine and to select/rank  $C$  is either the TF cosine or the TFIDF cosine. Results are reported for  $k = 100$  and  $\sigma = 0.05$ .

G-MAP	TF		TFIDF	
	LRT	SIM	LRT	SIM
NIPS	0.4489	0.3948	0.4531	0.4412
HEPTH	0.2432	0.2216	0.2543	0.2167



TFIDF  $C$  for LRT with  $k = 100$  and  $S = .05$

Figure 2.2: Precision vs. Recall on NIPS. The three lines are (from top to bottom) the LRT method's precision at a recall level with TFIDF cosine used to select  $C$ , the TFIDF distance  $C$  similarity baseline, and the TF distance  $C$  similarity baseline.

**LRTs are more accurate than similarities** Table 2.2 shows that the LRT achieves higher G-MAP scores than the similarity baselines on both NIPS and HEPTH. Among the two heuristic baselines, TFIDF cosine performs better than TF cosine. TFIDF cosine also appears to select better sets  $C$  for the LRT. The HEPTH results are reported for a random sample of 1600 documents.

Table 2.3: G-MAP scores comparing the LRT for a range of  $d^{(can)}$  influence mixing weights  $\sigma$  against the similarity baseline. The similarity measure to select  $C$  is either TF or TFIDF cosine. Results are reported on NIPS for  $k = 100$ .

G-MAP	TF		TFIDF	
	LRT	SIM	LRT	SIM
$\sigma = .001$	0.4575		0.4597	
$\sigma = .01$	0.4620		0.4649	
$\sigma = .05$	0.4489	0.3948	0.4531	0.4412
$\sigma = .1$	0.4475		0.4535	
$\sigma = .2$	0.4373		0.4447	

**LRT scores are more comparable than similarities** Table 2.2 showed that the LRT can find the most influential papers for one particular document. Figure 2.2 measures how well it can find the strongest edges in the whole influence graph. This precision-recall graph uses the ranking of all LRT statistic scores of all documents, with actual citations marked as positive examples. Figure 2.2 also shows the scores for using lists of TF and TFIDF cosine similarities. The LRT graph dominates the similarity baselines over the whole range and the difference in performance is larger than in the per-document evaluation. We conclude from this that LRT scores are more comparable between documents than similarity scores. This is to be expected because the LRT values have a clear probabilistic semantic. However, the similarity scores have no such guarantees.

**Effects of the  $\sigma$  parameter** Table 2.3 shows that the LRT is robust over a large range  $\sigma$  values. The LRT's G-MAP dominates the similarity baselines. However,  $\sigma = 0.01$  seems to perform better than our initial guess of 0.05 used above.

Table 2.4: G-MAP scores comparing the LRT against the similarity baseline for two  $k$ -NN approximation levels. The similarity measure for selecting  $C$  is either TF or TFIDF cosine. Results are reported on NIPS and HEPH for  $\sigma = .05$ .

G-MAP	TF		TFIDF	
	LRT	SIM	LRT	SIM
NIPS ( $k = 100$ )	0.4489	0.3948	0.4531	0.4412
NIPS ( $k = 10$ )	0.4067	0.3754	0.4580	0.4226
HEPTH ( $k = 100$ )	0.2432	0.2216	0.2543	0.2167
HEPTH ( $k = 20$ )	0.2227	0.2037	0.2264	0.1943

Table 2.5: How close is the approximation to the optimal? G-MAP scores are reported for  $S = .05$ .

Dataset ( $C$ )	GMAP	GMAP (perfect $C$ )
NIPS (TFIDF)	0.4531	0.4556
NIPS (TF)	0.4489	0.4590
HEPTH (TFIDF)	0.2543	0.3803
HEPTH (TF)	0.2432	0.3906

**Effect of  $k$  parameter in LRT approximations** Table 2.4 shows G-MAP scores at differing levels of the  $k$ -NN approximation. Recall from Table 2.2 that G-MAP scores for HEPH are substantially lower than for NIPS. We conjecture that this is due to the size of the corpus in relation to  $k$ . With a large corpus,  $k = 100$  is likely to exclude too many relevant documents from consideration. We further analyze the role of  $k$ , in its two roles in controlling the sizes of  $C$  and  $\mathcal{P}$ .

First,  $k$  controls the size of  $C$ . If  $k$  is too small, truly influential documents will not be tested by the LRT. E.g., in HEPH, each document has 14 citations on average. With



$k = 10$ , it would be simply impossible to recover the entire citation graph. Therefore we conclude that  $k$  must be large enough to include all documents that make contributions to  $d^{(new)}$ . On HEPH,  $k = 100$  is better than  $k = 20$  for TF and TFIDF cosine, and for LRT and similarity baseline. We believe this is because  $k = 20$  is too restrictive. NIPS with TF cosine shows the same behavior.

**Optimal  $C$**  To better understand how much loss in performance is due to the  $k$ -NN approximation of  $C$ , the following experiment explores the G-MAP scores of the LRT for a “perfect”  $C$ . In particular, we construct  $C$  so that it includes all documents that  $d^{(new)}$  actually cites, and then fill the remaining places in  $C$  with the most similar documents. Table 2.5 shows that for  $k = 100$  the loss in performance due to an approximate  $C$  is fairly small on NIPS. For HEPH, on the other hand,  $k = 100$  shows a much greater loss, with G-MAP scores only about 60-65% of the optimal. We believe this loss occurs because  $C$  is too small to accommodate all the influential documents.

### 2.4.3 Identifying Influential Documents

What are the influential documents that have the most effect on the document collection’s development? Which documents should one read to best grasp this development? We have already shown that LRTs can be used to infer an influence graph that is similar to a citation graph. We now investigate whether this influence graph can be used to identify the documents with the overall largest influence on the collection. In analogy to citation counts (i.e., the in-degree in the citation graph), we propose the in-degree in the influence graph as a measure of impact. If not noted otherwise, we form the influence graph by connecting each document  $d^{(new)}$  with the  $l$  other nodes that receive the highest LRT value. We typically use  $l = 10$ , although we also explore this parameter’s effect.

Table 2.6: The most influential paper per year in NIPS, as measured by influence graph in-degree, with  $k = 100$ ,  $\sigma = .05$ , and TFIDF cosine for  $C$ . We exclude years with edge effects and the last 3 years, since they are not statistically significant. Comparison is against the within-NIPS citation counts, and Google-scholar citation counts (on May 26, 2009).

Document		Citation Counts	
Year	Document Title and Author(s)	NIPS	Google Scholar
1988	“Efficient Parallel Learning Algorithms for Neural Networks” by Alan Kramer, A. Sangiovanni-Vincentelli	2	89
1989	“Training Stochastic Model Recognition Algorithms as Networks Can Lead to Maximum Mutual Information Estimation of Parameters” by John S. Bridle	11	172
1990	“Integrated Modeling and Control Based on Reinforcement Learning” by R. S. Sutton	0	44
1991	“Bayesian Model Comparison and Backprop Nets by David J. C. Mackay	1	38
1992	“Reinforcement Learning Applied to Linear Quadratic Regulation” by Steven J. Bradtke	6	73
1993	“Supervised Learning from Incomplete Data via an EM approach” by Zoubin Ghahramani, Michael I. Jordan	12	246
1994	“Reinforcement Learning Algorithm for Partially Observable Markov Decision Problems” by Tommi Jakkola, Sizarad Singhal, Michael I. Jordan	10	178
1995	“EM Optimization of Latent-Variable Density Models” by Chris M. Bishop, M. Svensen, Chistopher K.I. Williams	1	30
1996	“Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing” by V. Vapnik, Steven E. Golowich, Alex Smola	2	610 (13364)
1997	“EM Algorithms for PCA and SPCA” by Sam Roweis	1	267

### Qualitative evaluation

For each year in NIPS, Table 2.6 lists the paper with the highest in-degree in the influence graph computed by the LRT method with  $k = 100$  and  $l = 10$ . We expect these

Table 2.7: Rank metrics comparing the LRT against similarity on NIPS ( $k = 100$ ) and HEPH ( $k = 20$ ), using  $\sigma = .05$  and TF or TFIDF cosine for  $C$ . We ignore the first two and last two years because of edge effects.

Corpus	TF					
	LRT			SIM		
	$\tau$	RMap@3	@12	$\tau$	RMap@3	@12
NIPS	0.4216	0.2771	0.3126	0.3379	0.1475	0.2561
HEPTH	0.3887	0.2558	0.2376	0.3497	0.1421	0.1594

Corpus	TFIDF					
	LRT			SIM		
	$\tau$	RMap@3	@12	$\tau$	RMap@3	@12
NIPS	0.4163	0.2751	0.3022	0.3686	0.1959	0.2585
HEPTH	0.3549	0.1456	0.1582	0.3190	0.1139	0.1138

to have high citation counts, which we test by showing the paper’s citation counts both from within the NIPS corpus (as of 2000) and from Google Scholar (as of 2007). For most documents, the citation count is indeed high when compared to the average NIPS document citation count of 0.7734 other NIPS papers. An interesting example is “Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing” from 1996. While this is one of the papers that introduced SVMs to NIPS, it has only 3 citations within NIPS and only 44 citations in Google Scholar. Nevertheless, SVMs had a huge impact on NIPS. In this sense our LRT method is correct and is not influenced by citation habits. In this example, most authors cite Vapnik’s later book (with 5144 citations) instead of this paper. The LRT method is unaffected and correctly identifies the SVM idea as highly influential on NIPS.

## Quantitative Evaluation

We compare the ranking of documents by in-degree in the influence graph to the ranking by citation count. As similarity measures, we use Kendall’s  $\tau$  and a ranking version of MAP, which we term R-MAP.

**Kendall’s  $\tau$**  Kendall’s  $\tau$  measures how many pairs two rankings rank in the same order. It ranges between -1 and 1, with higher numbers indicating greater similarity. Formally,

$$\tau = \frac{2 \cdot \text{number of concordant pairs}}{\text{total number of pairs} - \text{number of tied pairs}}$$

**R-MAP@ $k$**  R-MAP@ $k$  measures the average precision of a ranking. With the  $k$  top-ranked documents as positive examples, average the ranking’s precision at the positions of these documents. We calculate R-MAP@3 and R-MAP@12.

There is one caveat with rank-based metrics. Edge effects (e.g., older papers have more citations, papers from the last year have no citations) make it difficult to present one unified ranking of all documents. Therefore, we calculate each metric per-year and average the year-by-year values to get a single score for the entire corpus. Additionally, because of edge effects, the first two and the last two years are not used, since they do not contain meaningful results.

The TF and TFIDF baselines use the most similar documents instead of the LRT predictions.

**LRTs are better than similarity** Table 2.7 shows that the LRT gives substantially better rankings than the similarity baseline for all metrics on both NIPS and HEPATH with both TF and TFIDF cosine  $C$ .

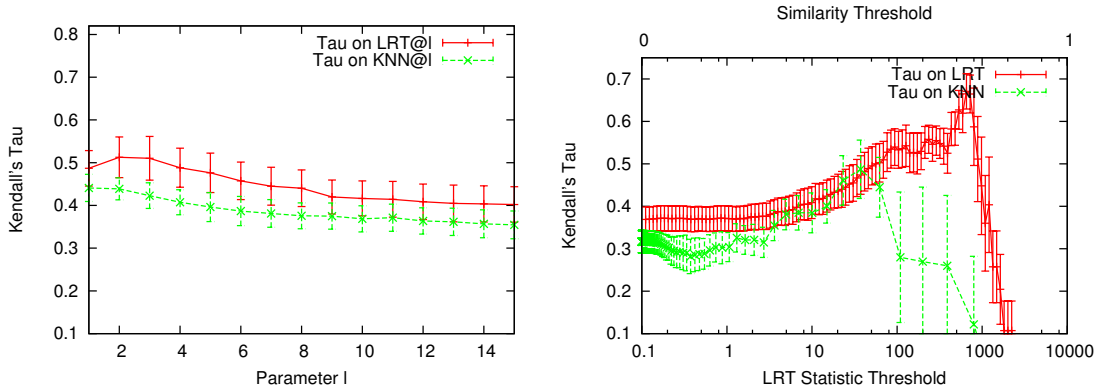


Figure 2.3: Using  $\tau$  to compare the LRT against the similarity baseline, both with the  $l$  parameter (left) and by thresholding the LRT statistic values (right). Results are for NIPS with TFIDF cosine  $C$  and  $k = 100$ . The TF plot looks similar, except that the baseline is smoother.

**Effect of the parameter  $l$**  The left plot of Figure 2.3 explores whether selecting influencers is sensitive to the parameter  $l$ . For the influence graph, we considered each document's  $l$  predicted influencers with highest LRT scores. Figure 2.3 shows how varying  $l$  affects  $\tau$  for both LRT and the similarity baseline. Since NIPS documents do not have many citations, we explore  $l = 1$  to 15. The upper line is LRT performance with 95% confidence interval error bars. (The confidence interval is computed using the multiple  $\tau$  values per data point, because each graphed  $\tau$  is the average of multiple (here, 10) years of  $\tau$  metric scores.) The lower line depicts  $\tau$  on the similarity baseline. For the TFIDF cosine  $C$ , when  $l$  is small, the method computes a count over only the few top influential documents selected by the LRTs for  $d^{(new)}$ . It turns out that small  $l$  seem to perform better than our initial guess of  $l = 10$ . As  $l$  increases, more non-influential documents are counted and  $\tau$  correspondingly falls. When  $l$  approaches 100 (not shown), the LRT and the baseline are identical as expected by construction.

**Thresholding on the LRT score** The right plot of Figure 2.3 depicts how  $\tau$  varies if we do not select a fixed number of  $l$  neighbors per document, but instead use a threshold

on the LRT statistic. The LRT is set up to reject the null hypothesis and declare that  $d^{(can)}$  influences  $d^{(new)}$  if the LRT statistic is sufficiently large. Varying this threshold controls the level of confidence in the LRT, so we use the threshold level as the x-axis and examine how it affects  $\tau$ . Thresholding the LRT values actually gives better performance than using the  $l$  parameter, since we are not forcing a certain number of influence links for each document. There are four different regions in this graph. First, if the threshold is too low, performance suffers because the null hypothesis is being accepted erroneously. Second, performance increases as the threshold approaches reasonable confidence levels. Third, a large range of threshold values (approximately 100-2000) give good and similar  $\tau$  scores, showing that the LRT method is robust. Fourth, when the threshold is too high, many influential documents are no longer detected, and performance subsequently falls.

Note that a confidence level of 95% per test (i.e., a threshold of 3.84) performs quite poorly. This level means that 5% of the influence links are erroneous. NIPS, with 2000 papers, would have an expected 100,050 false links (and only 1512 real citations). Therefore, we need a much higher confidence level to account for the multiple-testing bias. Using Bonferroni adjustment, each test's level is the overall level divided by the number of tests.

## 2.5 Discussion and Future Work

One obvious limitation of the current model is the simplicity of the language model. The assumption that each document is a sequence of independent words is, in reality, clearly violated. This observation motivates more expressive language models such as  $n$ -gram language models.

There is also the question of whether these methods can generalize to other domains. LRTs do not use citation data, so many domains should be applicable. However, we have only conducted experiments on research publications.

Finally, there is scalability and efficiency. Much of the computing time is spent solving convex optimization problems. While  $C$  and  $\mathcal{P}$  prune this space, there may be other criteria to provably eliminate certain LRTs without affecting the results. Furthermore, the optimization problems have a special structure, which can probably be exploited by specialized methods to solve the optimization problems.

## 2.6 Summary

We presented a probabilistic model of influence between documents for corpora that have grown over time. In this model, we derived a Likelihood Ratio Test to detect influence based on the content of documents and showed how the test can be computed efficiently. We found that the influence graphs derived from the content resemble the structure of explicit citation graphs for corpora of scientific literature. Furthermore, we showed that in-degree in the influence graph is an effective indicator of a document's impact. The ability to create influence graphs based on document content alone has the potential to open databases without explicit citation structure to the large repertoire of graph mining algorithms.

## CHAPTER 3

### NOVEL IDEAS IN TEXT DOCUMENTS

The Inter-Document Influence Model can identify how ideas flow between documents in corpora and provides a text-based method for finding influential documents and their ideas. Keeping in mind the high-level goal of understanding the idea structure in the corpus, one logical next step is to find the places where novel ideas in the corpus originate. The task of detecting novel ideas is interesting because it focuses on what ideas make a document new or different with respect to existing documents and their ideas. The novel ideas are the ones that push the boundaries of the content expressed in the document collection. The hope is that novelty detection methods can identify new and interesting advances in the content that the corpus covers.

#### **3.1 Introduction**

Novelty detection has formerly been addressed in the literature, including as a task in the well-known Topic Detection and Tracking (TDT) studies (Allan et al., 1998a). Novelty detection in the TDT studies means marking news articles that cover new events or topics. In this thesis, the novelty detection task differs in the following two ways: detecting novelty with respect to earlier documents instead of real-world newsworthy events, and offline analysis of the entire collection at once instead of emphasizing online novelty detection. Besides the TDT studies, TREC has also organized several novelty tracks, where the task was to mark the sentences in a ranked set of search results that provide novel information. In that work, novelty was defined with respect to new information about the user's query in a list of retrieved search results, so as to reduce redundancy in the results. This thesis considers novelty with respect to time, finding new ideas in



documents as compared to older documents.

In the archival corpus setting considered in this thesis, addressing novelty means identifying both the new ideas in the corpus and the documents that introduced those ideas. Obvious application domains for such a novelty detection system are news articles and research literature. For news articles, scoring documents by their novelty might provide a ranking of the first articles that broke important stories. For research publications, identifying the documents that contain the most novel content is also quite interesting. Since the research literature is expected to be highly novel, applying novelty detection methods to finding the most different ideas in such a setting is potentially very exciting. We therefore will use research publications for experimental evaluation.

We propose two specific methods for the text-based novelty detection task. The first method makes a description for each document's novel idea by listing the set of terms that are most novel in that document as compared against the background of ideas presented in previous documents. The most novel words are those which have the most increased likelihood of occurring in the document vs. occurring in the ideas from previous documents. For this method, we present qualitative results on the most novel terms for selected documents in a set of research publications. Besides describing what makes each document novel, another interesting task is to determine just how novel each document is. The second novelty detection method quantifies the amount of novelty per document. This method is an application of information theory, in particular the KL-Divergence, which arises straightforwardly as a scoring function to quantify document novelty. We evaluate these methods on real and synthetic data from the research publications that appeared in the NIPS conference (NIPS Online, 2000).

## **3.2 Related Work: Novelty Detection**

There are various existing approaches for the task of detecting novelty in various settings. The novelty detection task as we set it up here differs somewhat from these existing approaches.

### **3.2.1 Novelty Detection in TDT**

In the Topic Detection and Tracking studies (Allan et al., 1998a; Allan et al., 1998b), the novelty detection task focused on identifying the news articles that first introduced novel topics. This task specifically focused on novel topics with respect to news events. The desired output of this method is a set of articles that mark the first instances where each event is mentioned.

The novelty detection task in the TDT setting differs from our notion of identifying novel documents in several ways. First, the TDT studies place emphasis on the online detection of novel articles in a news stream, while our methods focus more on retrospective analysis of the document collection. Second, the focus in the TDT studies is on novelty with respect to real-world newsworthy events, while we define novelty with respect to ideas as they are represented in text. For example, in the TDT3 corpus, the news articles cover two different hurricanes. In the novelty event detection framework, these are two different events, and thus the first documents reporting on each hurricane should be marked as novel. With regard to ideas and their manifestation in text, however, the “ideas” presented in the articles are probably very similar. The earliest coverage of the first hurricane should still be detected as novel, but unless there was significantly different circumstances, analysis, or reporting on the second hurricane, the text content

presumably would not be very novel. This also means that our methods do not use a “window” or “forget” past events. Especially for research literature, we really do want to compare novelty against the text of all existing ideas, not just some recent window, while in the TDT novelty detection setting, some window to forget older content is essential.

The information theoretic method that arises from our document collection assumptions scores documents based on KL-Divergence. KL-Divergence has also been used in the TDT setting (Lavrenko et al., 2002).

### **3.2.2 TREC Novelty Track**

Besides the TDT studies, there have been other analyses of novelty in data mining. Another well-known group of work is the Novelty Track from various TREC conferences, e.g., (Soboroff & Harman, 2003). Even though that task would appear to be quite similar to this one in name, in fact they are really quite different. The TREC Novelty Track’s task focuses on the setting of information retrieval, where there are a list of retrieved documents for a query. Some of the sentences in the retrieved document set have sentences that are relevant to the user’s information need. As the user would read through the list of retrieved documents, the novel sentences consist of the subset of the relevant sentences that contain new information related to the user’s query. Here, we detect novelty in the context of a time-sorted, archived document collection, instead of a set of search results. Furthermore, the representations of novelty that we consider, in novel words and scores for the degree of document novelty, differ from the output of the novel relevant sentences.

### **3.2.3 Other Novelty Tasks**

There are other settings where novelty has been considered in the literature, especially when considering novelty to mean different. Novelty is really a combination of being different and occurring later in time. Work from language modeling and temporal document clustering have implicitly considered issues relating to novelty (Blei & Lafferty, 2006; Mei & Zhai, 2005).

## **3.3 Task 1: Describing Novel Ideas in Their Own Words**

One goal of novelty detection is to help people understand the novel ideas in a document collection. Here, we present a method that summarizes each document's novel idea as a list of the document's most novel terms. Given a document, it will extract what is novel about that document. The hope is that such a method can help users understand what is novel about each document.

### **3.3.1 Method**

In the context of the mixture models from Ch. 2, the most obvious way to identify novel content is to use the probabilities from the estimated novel language model in the Inter-Document Influence Model with Novel Content (Model 2.3). Even though this model was explicitly designed for influence, not novelty, one of the side-effects is that it contains a description of the novel content of each document. Even though this model may not be the most straightforward for novelty detection, it provides a springboard for our analysis of novelty. As before, we have the generative assumption that each

document's text is drawn from a mixture of a novel language model and the language models estimated from prior documents.

Assume that each document  $D^{(i)}$  is a vector of  $n_i$  words  $W^{(i)} = (W_1^{(i)} \cdots W_{n_i}^{(i)})'$ , which come from a vocabulary  $V$ . As in the influence model, assume that documents are generated from a mixture of language models  $\hat{\theta}^{(k)}$  derived from a set of already-existing previous documents  $D^{(k)}$ , as well as a document-specific novel language model  $\bar{\theta}^{(i)}$ . The mixing weights  $\pi^{(i)}$  are denoted by  $(\pi_n^{(i)}, \pi_k^{(i)})$  for  $\bar{\theta}^{(i)}$  and  $\hat{\theta}^{(k)}$ , respectively. With these definitions and assumptions,

$$P(d^{(i)} | \pi^{(i)}, d^{(1)} \cdots d^{(i-1)}) = \prod_{j=1}^{n_i} (\pi_n^{(i)} \bar{\theta}_{w_j^{(i)}}^{(i)} + \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_{w_j^{(i)}}^{(k)})$$

The combination of previous-document unigram language models comprises the background mixture against which the method will try to identify the document's novel ideas. According to this model, the entire document's content consists of words drawn from this background mixture and words drawn from the unigram multinomial distribution for novel content.

## Inference

With this generative model, we can use the inferred novel language model to find the most novel terms in the document. When fitting the model to explain the content of a document  $d^{(i)}$ , the observed quantities are the text of all the documents. The parameters are the mixture weights  $\pi^{(i)}$  for document  $d^{(i)}$  and the probabilities in the novel language model  $\bar{\theta}^{(i)}$ . Inference should select the maximum-likelihood parameters for generating the content of document  $d^{(i)}$ .

As it stands, this inference problem cannot be optimized globally in a straightfor-

ward manner. To make the likelihood function convex, one solution is to constrain the document to have a certain constant amount of novel content, in effect setting the  $\pi_n^{(i)}$  to a fixed value. This may not necessarily be a bad assumption. Research publications, for example, are supposed to have a large amount of novel content, while typical news articles might have several paragraphs (i.e., a relatively similar amount) of novel updates followed by some background on the story. Once we have chosen a value for this parameter, solving the maximum likelihood problem provides an estimated distribution for the novel language model  $\bar{\theta}^{(i)}$  and mixing weights for the background mixture. The most novel terms are those with the highest probability in the novel language model relative to the background mixture. Therefore, we rank the terms by their probability in the inferred novel language model minus the probability in the background mixture. We select the top terms from this ranking as the most novel terms for document  $d^{(i)}$ .

### 3.3.2 Experiments

We evaluate this method on the NIPS collection of fulltext research publications from the Neural Information Processing Systems conference (NIPS Online, 2000), both on the real data and on synthetic data that was generated based on these documents.

#### Novel Terms in Synthetically-Generated Data

To evaluate this method, we directly measure how well the learned novel language model represents the actual novel language model used to generate the data. Here, we generated the synthetic documents  $d^{(i)}$  according to Model 2.3, with novel language models being maximum likelihood estimators of NIPS documents. First a set of  $k_P = 10(100)$  previous document indices  $\mathcal{P}$  is selected uniformly at random which will be used for

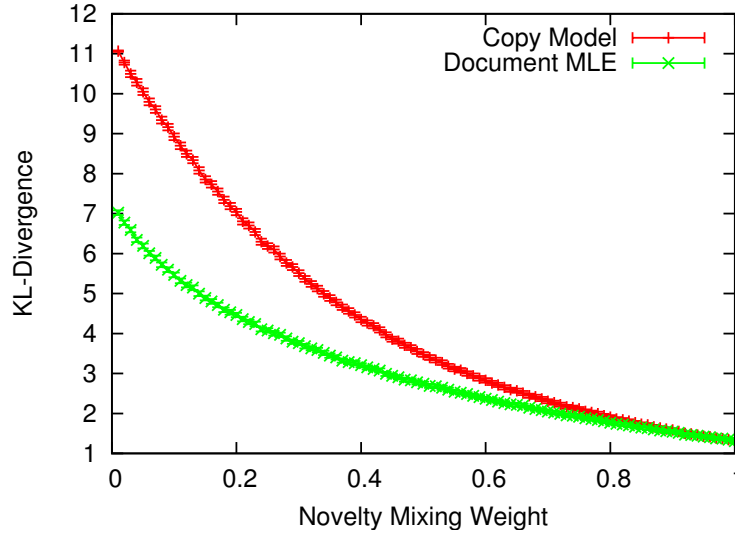


Figure 3.1: KL-divergence from the actual novel content distribution to the novel distribution learned according to the Inter-Document Influence Model (Model 2.3). The baseline shown is the KL-divergence from the actual novel language model to the MLE from the entire generated document. The x-axis is  $\pi_n^{(i)}$ . At a  $\pi_n^{(i)}$  level, the KL-divergence measures the amount of extra bits the inferred original content distribution (and baseline) need to encode the information in the true original content distribution.

the  $k_P$  previous documents  $d^{(1)} \dots d^{(k_P)}$ . Then, a novel language model  $\bar{\theta}^{(i)}$  (for  $i > k_P$ ) is chosen by using the MLE of another NIPS document selected uniformly at random. There are 101 documents that are generated using these models so that they have novel content mixing weights  $\pi_n^{(i)}$  of  $\{0, 0.01, 0.02, \dots, 1\}$ . The other mixing weights  $\pi_k^{(i)}$  are selected uniformly at random and normalized to sum to  $1 - \pi_n^{(i)}$ . This entire process was repeated 100 times, for new selections of  $(\mathcal{P}, \bar{\theta}^{(i)}, \pi^{(i)})$ . The words in documents are drawn according to their document language models, with the document length set to 1400 words, since that is the average NIPS document length.

We fit the Inter-Document Influence Model to these generated documents to estimate the novel language model of each document  $d^{(i)}$ , for  $i > k_P$ . To measure the quality of the model for detecting novel terms, we measure how well this learned novel language

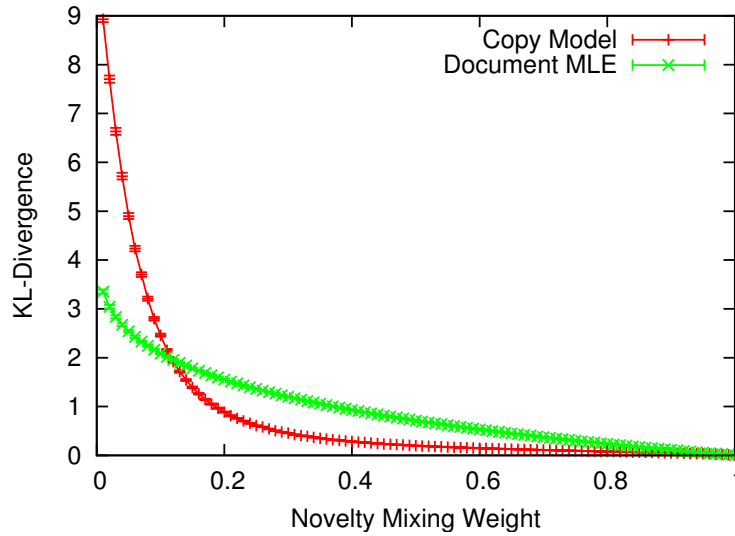


Figure 3.2: KL-divergence from the actual novel content distribution to the learned novel distribution. The baseline is the KL-divergence from the actual novel language model to the MLE from the entire generated document. The generated document length was 100000 words.

model captures the actual novel language model used to generate the documents. We quantify this evaluation by using the KL-divergence of the learned novel language model from the actual novel language model. As a baseline, we used the entire document as an approximation of the novel content. In this case, the baseline value is the KL-divergence of the entire generated document from the actual novel language model. For both methods, we smoothed both distributions in the KL-divergence computation with Jelinek-Mercer smoothing with  $\lambda = 0.001$ .

Figure 3.1 shows this comparison, where the generated document baseline actually does much better than the novel terms method. This shows that when considering the whole distribution, the generated document is a better approximation of the novel language model than the inferred novel language model from the copy model. One conjecture is that the copy model places into the novel language model the terms that just happened to be drawn more frequently, even though they are from the background, es-



pecially if they had a high probability in the background. To test whether this was the case, we repeated the experiment with documents of length 100000 words. The longer documents mean that the drawn documents from the distributions should more accurately reflect those distributions. Figure 3.2 shows that in this case the KL-Divergence for the learned novel language model does outperform the generated document baseline when there is 10% or more novel content. With very small amounts of novel content, however, the generated document is better. To understand the reason for this, we read lists of the highest probability terms from the learned novel language model when there is little novel content and found that there were terms in the lists that were associated with high probabilities in the previous documents. The intuition is that it is better for the model to “fix” the probabilities for high-probability background terms that were drawn more often randomly than to adjust low-probability words actually drawn from the novel language model.

### **Novel Terms in Influential Documents**

Next we perform a qualitative evaluation, identifying the most novel terms in the most influential documents according to Model 2.3. This set of most influential documents was presented in Table 2.6. Here, we summarize the set of novel terms by presenting the terms that have the most difference in probability between the novel language model and the background mixture of previous language models. As a baseline, we present the list of terms that have the highest TFIDF values for each document. These results are shown in Table 3.1. The novel terms selected by the Influence Model and the highest-weighted TFIDF terms are quite similar for many documents. In the earlier documents, the TFIDF terms seem to be somewhat more relevant to the novel aspects of each document’s subject. Perhaps this is because TFIDF has the advantage of looking into the future, since

Table 3.1: Top 10 novel terms and highest-TFIDF (similarity baseline) terms for the yearly most-influential NIPS paper (Papers from Table 2.6). With  $k_P = 100$  and  $\pi_n^{(i)} = 0.05$ .

Document Title and Author(s), Copy Model Novel Terms, Top-TFIDF Terms
1990 “Integrated Modeling and Control Based on Reinforcement Learning” by R. S. Sutton Nov: dyna, planning, ahc, world, hypothetical, reward, architectures, trial, maze, experience Sim: dyna, ahc, planning, policy, world, sutton, hypothetical, reinforcement, reward, ...
1991 “Bayesian Model Comparison and Backprop Nets” by David J. C. Mackay Nov: evidence, occam, comparison, mackay, bars, ed, factor, posterior, aw, razor Sim: occam, evidence, bayesian, gull, razor, diw, inference, interpolant, mackay, ...
1992 “Reinforcement Learning Applied to Linear Quadratic Regulation” by Steven J. Bradtke Nov: lqr, linear, quadratic, iteration, ut, bradtke, regulation, zt, qv, qt Sim: lqr, policy, dp, bradtke, controller, watkins, reinforcement, qv, ut, regulation
1993 “Supervised Learning from Incomplete Data via an EM approach” by Zoubin Ghahramani, Michael I. Jordan Nov: em, incomplete, expectation, hij, xii, gaussians, maximization, valued, involve, lse Sim: missing, mixture, density, incomplete, em, hij, olok, xii, lse, xi
1994 “Reinforcement Learning Algorithm for Partially Observable Markov Decision Problems” by Tommi Jakkola, Sizarad Singhal, Michael I. Jordan Nov: reward, alm, pomdp, slm, messages, message, improvement, average, mdp, learner Sim: policy, alm, slm, mdp, pomdp, reward, learner, policies, messages, markov
1995 “EM Optimization of Latent-Variable Density Models” by Chris M. Bishop, M. Svensen, Chistopher K.I. Williams Nov: latent, oil, matrix, williams, visualization, svens, pipe, toy, gas, elements Sim: latent, oil, em, pipe, svens, bishop, variable, visualization, density, distribution
1996 “Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing” by V. Vapnik, Steven E. Golowich, Alex Smola Nov: sv, svcs, xi, inner, solving, xj, hilbert, smola, golowich, estimation Sim: svcs, sv, splines, xi, xj, golowich, inner, smola, kernel, hilbert
1997 “EM Algorithms for PCA and SPCA” by Sam Roweis Nov: space, covariance, cc, datapoints, subspace, iterations, guess, rod, xly, panel Sim: spca, pca, covariance, principal, em, subspace, datapoints, eigenvectors, cc, missing

it is computed over the whole corpus. In later documents, the method and the baseline are more similar. One thing to notice is that both methods seem to pick up on the most unusual terms, as can be seen in the most novel or highest TFIDF terms containing

notation from the math or the author names from the document headings. This shows that novelty alone is somewhat tricky, because although these terms are indeed novel, they are not useful for helping a human understand the development of that document's ideas. Overall, this method was interesting to try, but does not appear qualitatively to be much better than simply using TFIDF to select words.

### **3.4 Task 2: Quantifying Each Document's Novelty**

The next novelty task is motivated by the application of identifying the most novel documents. Here, we present a method to quantify the amount of novelty in each document.

#### **3.4.1 Method**

The method for novel terms came about as a side effect of the Inter-Document Influence Model (Model 2.3). That method had to assume a constant amount of novel content per document, which is quite likely violated in practice. We now present a completely different method for novelty with the goal of quantifying the amount of novel content in each document. To measure the amount of novelty per document more precisely, we propose an information theoretic method to score documents by novelty. Detecting novel documents can be re-framed as identifying the documents whose content most differs from existing content in previous documents. The more novel a document's ideas are, the more difficult it would be to explain that document's content with the word distributions arising from the previous documents. We propose a method to quantify novelty in documents by using the Kullback-Leibler Divergence (KL-Divergence) ( $D_{KL}$ ).

## KL-Divergence Method

The goal of the novelty task is to quantify how novel each document's content is. The score for each document  $d^{(i)}$  should be based on how much extra novel information  $d^{(i)}$  has when compared to the background of existing documents. For this task, we use KL-Divergence because it measures the number of extra bits that are needed to encode the novel unigram distribution when using an approximation for that distribution. To apply the KL-Divergence to this novelty task, each document  $d^{(i)}$  should be scored by how much extra information is needed to encode the content of  $d^{(i)}$  by using a code based on best linear combination of prior existing documents  $d^{(1)} \dots d^{(i-1)}$ . This scoring function obeys the property we want, namely that if  $d^{(i)}$  has a large amount of novel content, then the distribution of  $\hat{\theta}^{(i)}$  will differ quite a bit from the mixture defined by  $\pi^{(i*)}$ . This will require more extra bits in the representation, which is equivalent to producing a larger novelty score.

We use the same notation, except with a slight change in the generative model. Here, we assume that documents are generated from a mixture of existing language models only, without any novel language model. The previous document language models are still parameterized by  $\hat{\theta}^{(k)}$ . Their mixing weights  $\pi^{(i)}$  consist of  $\pi_k^{(i)}$  for the estimated distributions  $\hat{\theta}^{(k)}$ , but no longer a novel mixing weight  $\pi_n^{(i)}$ . With these assumptions, we have this generative probability for a document

$$P(d^{(i)}|\pi^{(i)}, d^{(1)} \dots d^{(i-1)}) = \prod_{j=1}^{n_i} \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_{w_j^{(i)}}^{(k)}$$

The inference method that arises from this model is based on seeing how well the copy model can explain the content of document  $d^{(i)}$  using only existing ideas. Specifically, it tries to explain the content of  $d^{(i)}$  using only a mixture of existing documents, without any novel language model. To use KL-Divergence for this task, we want to find

the divergence of the observed  $w^{(i)}$  from the best possible explanation for the words  $w^{(i)}$  by using the language models  $\hat{\theta}^{(1)} \dots \hat{\theta}^{(i-1)}$  from the documents that precede  $d^{(i)}$ . In the following, we overload the notation  $\hat{\theta}^{(i)}$  to refer additionally to the distribution parameterized by  $\hat{\theta}^{(i)}$ , where appropriate. The optimal mixing weights are given by

$$\begin{aligned}
\pi^{(i*)} &= \operatorname{argmin}_{\pi^{(i)}} KL(\hat{\theta}^{(i)} \parallel \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_{w_j^{(i)}}^{(k)}) \\
&= \operatorname{argmin}_{\pi^{(i)}} - \sum_{w \in V} \hat{\theta}_w^{(i)} \log \left( \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_w^{(k)} \right) + \sum_{w \in V} \hat{\theta}_w^{(i)} \log \hat{\theta}_w^{(i)} \\
&= \operatorname{argmin}_{\pi^{(i)}} - \sum_{w \in V} \hat{\theta}_w^{(i)} \log \left( \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_w^{(k)} \right)
\end{aligned}$$

subject to  $\pi_n^{(i)} = 0$  in the minimization, and the novelty score for  $d^{(i)}$  is given by

$$s_n^{(i)} = - \sum_{w \in V} \hat{\theta}_w^{(i)} \log \left( \sum_{k=1}^{i-1} \pi_k^{(i*)} \hat{\theta}_w^{(k)} \right) + \sum_{w \in V} \hat{\theta}_w^{(i)} \log \hat{\theta}_w^{(i)}$$

Since the KL-Divergence is a convex function, this is straightforwardly optimizable.

We use the optimization package MOSEK (MOSEK, 2008) to do the optimization.

## Handling Zero Probabilities

In the KL-Divergence computation, since document text is sparse, many terms will be associated with a probability of 0 in the maximum likelihood estimates of unigram multinomial distributions derived from documents in the corpus. One way to resolve this situation is to smooth the term probabilities. We explore several common ways of smoothing the term probabilities, including Jelinek-Mercer smoothing, Dirichlet smoothing, and discount smoothing. These smoothing methods and their application to information retrieval tasks have been explained previously (Zhai & Lafferty, 2004), and we will mostly follow their notation and setup for smoothing in this section.

For each smoothing method, the word probabilities from the maximum likelihood unigram language models estimated directly from each document's words will be smoothed by the probabilities of those words occurring in the corpus. Let  $\hat{\theta}_w^{(\mathcal{D})}$  represent the maximum likelihood probability of word  $w$  in the corpus  $\mathcal{D}$ . Then, when smoothing the unigram distribution  $\hat{\theta}^{(k)}$  from document  $d^{(k)}$ , the probability of a word  $\hat{\theta}_w^{(k)}$  will include a summand  $\alpha_k \hat{\theta}_w^{(\mathcal{D})}$ .

**Jelinek-Mercer Smoothing** In Jelinek-Mercer smoothing, the  $\alpha_k$  is constant across all documents  $d^{(k)}$ . This smoothing amount is typically referred to by the parameter  $\lambda = \alpha_k$ . Instead of  $\hat{\theta}_w^{(k)}$ , the probability of a word  $w$  in the language model derived from a previous document  $d^{(k)}$  is

$$\lambda \hat{\theta}_w^{(\mathcal{D})} + (1 - \lambda) \hat{\theta}_w^{(k)}.$$

Working through the algebra, this problem is straightforwardly optimizable using a similar setup as the optimization problem without smoothing, except that some quantities are scaled by  $\lambda$  or  $1 - \lambda$ .

**Dirichlet Smoothing** Dirichlet smoothing is often described in terms of a parameter  $\mu$ , which is the (uniform) number of a priori counts that each word is assumed to have. As a special case, when  $\mu = 1$ , the smoothing method is called Laplace smoothing. In this case, the smoothing parameter  $\alpha_k = \frac{\mu}{|d^{(k)}| + \mu}$  depends on the length of document  $d^{(k)}$ . The probability of a word  $w$  in the language model for a previous document  $d^{(k)}$  is now given by

$$\alpha_k \hat{\theta}_w^{(\mathcal{D})} + (1 - \alpha_k) \hat{\theta}_w^{(k)},$$

which looks very similar to the Jelinek-Mercer smoothing, except that  $\alpha_k$  is not constant across documents with Dirichlet smoothing. By introducing one more variable

and working through some details, the optimization problem can once again be written as a separable convex program. We introduced an extra variable to preserve sparsity in the linear constraint matrix for obvious space efficiency reasons.

**Discount Smoothing** Discount smoothing differs from Jelinek-Mercer and Dirichlet smoothing in that probability mass is removed (discounted) from the higher probability terms and added onto the lower probability terms, instead of smoothing only by adding some a priori term counts. The typical parameterization for Discount smoothing uses the parameter  $\delta \in [0, 1]$ , which represents the number of the term counts that are removed from the higher probability terms. In this case, the parameter  $\alpha_k = \frac{\delta |d^{(k)}|_u}{|d^{(k)}|}$ , where  $|d^{(k)}|_u$  is the number of unique terms that are present in document  $d^{(k)}$ . The probability of a word  $w$  in the language model for  $d^{(k)}$  can be written as

$$\alpha_k \hat{\theta}_w^{(D)} + \max(\hat{\theta}_w^{(k)} - \frac{\delta}{|d^{(k)}|}, 0)$$

Once again, with appropriate rewriting, this is straightforwardly optimizable as a separable convex program, with only one non-sparse vector for the background smoothing probabilities in the constraint matrix.

## Implementation Details

As in the Influence Model optimizations, we restrict the set of previous documents to be the most similar  $k_p$  documents according to cosine similarity. This approximation has the effect of making the optimizations run much faster, while sacrificing some possible previous documents and their ideas. Although we do not evaluate this approximation here, we expect that in practice, it does not make a large difference, as was also the case with the Influence method. When two documents have an influence relationship,

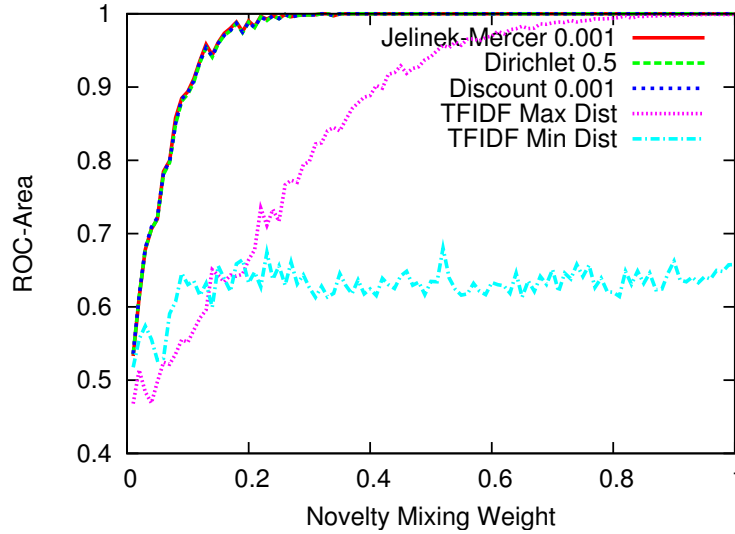


Figure 3.3: ROC-Area analysis of the novelty score. The x-axis shows the amount of novel content  $\pi_n^{(i)}$  in the generated documents. The graph shows ROC-Area of points ordered by the novelty score, for the various smoothing methods, with documents at this  $\pi_n^{(i)}$  level being positive points and generated documents with  $\pi_n^{(i)} = 0$  being negative points. The TFIDF baseline is based on taking the cosine distance from document  $d^{(i)}$  to the single nearest document that precedes it in time.

the influenced document draws on the ideas from the influencing document. One would expect that these documents have some level of similarity, namely in the text regarding the idea they share in common. Here, we expect that the approximation is valid because if the document is novel with respect to the documents with the most similar content, it is likely to be even more novel when compared against other more dissimilar documents.

### 3.4.2 Experiments

We test the method for quantifying the amount of novel content per document by using synthetic data based on research publications from NIPS. Then, we present qualitative results for actual NIPS publications.



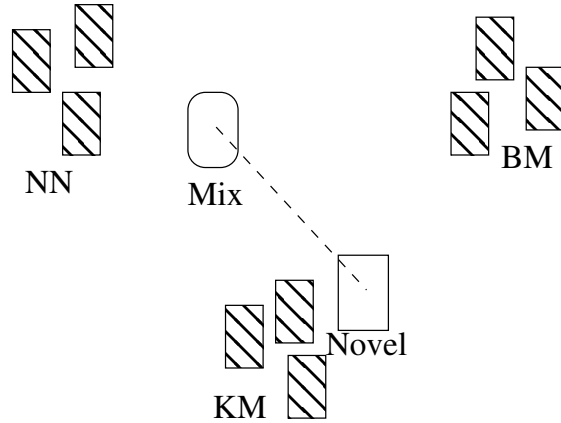


Figure 3.4: This figure depicts the TFIDF Min distance baseline and why it may not work in the setting in which we generate documents. The groups of shaded boxes represent the MLEs from previous documents of similar topics, perhaps Neural Networks, Bayesian Methods, and Kernel Methods. The rounded box marked Mix in the middle represents the weighted mixture of the previous documents. The box marked Novel represents the novel language model. Here, as the novelty weight is increased in the generated documents, the generated documents will tend to lie along along the dotted line reaching from Mix to Novel. When Novel is similar to a previous document, the cosine distance between the TFIDF vectors of the generated document and the closest previous document may actually decrease with increasing novelty instead of increasing as would be intuitive.

### Novelty in Synthetically-Generated Data

First, we evaluate the novelty score  $s_n^{(i)}$  on synthetic data that was generated based on the maximum-likelihood estimators from NIPS documents. Here, we generate the data according to the Inter-Document Influence Model as described in Section 3.3.2.

We conduct the evaluation of the novelty score on this generated data using ROC-Area in the following manner. For each positive fraction of novel content  $\pi_n^{(i)}$ , we treat the novelty scores of the documents  $d^{(i)}$  with that  $\pi_n^{(i)}$  as positive points, while the novelty scores of the documents generated with  $\pi_n^{(i)} = 0$  are negative points. The novelty score is used to order these points. At each level of  $\pi_n^{(i)}$ , we compute the ROC-Area.

We use two baselines based on TFIDF. Since TFIDF was designed to find the terms that most represent a document relative to the rest of the corpus, it should be an effective method for measuring what is novel about a document with respect to the past. The documents used to compute the IDF values are the background documents  $d^{(k)}$  as well as the single  $d^{(i)}$  at the specified novelty level  $\pi_n^{(i)}$ . Since only one document in this set has novel content, TFIDF should emphasize that document's most novel terms. As baselines, we compute for each document the cosine distance from the single nearest previous document (TFIDF Min) and the cosine distance from the single farthest previous document (TFIDF Max). The Min/Max refers to the minimum or maximum amount of novel content a document could have when compared against any single document from the past.

Figure 3.3 presents the results of this evaluation. In fact, with optimal parameter settings, all three smoothing methods with the Influence Model seem to do quite similarly. If the documents have 20% novel content or more, the method is able to distinguish it from the documents that do not contain any novel content with almost perfect accuracy. With very little novel content, e.g.,  $\pi_n^{(i)} = 0.01$ , all methods and baselines have an ROC-Area near 0.5, which would be random. Both TFIDF baselines do much worse than the KL-Divergence method.

Interestingly, TFIDF Max (distance) does much better than TFIDF Min (distance), which seems counterintuitive. Thinking of the language models as vectors, the MLE of the novel document is in fact just another vector. Because of the way we generated the documents, it likely shares quite a few words in common with some other background documents. E.g., as shown in Figure 3.4, with 100 previous documents, perhaps there are 40 on neural networks, 30 on Bayesian methods, and 30 on kernel methods. Now, if the novel language model happened to be chosen as the MLE of another kernel meth-

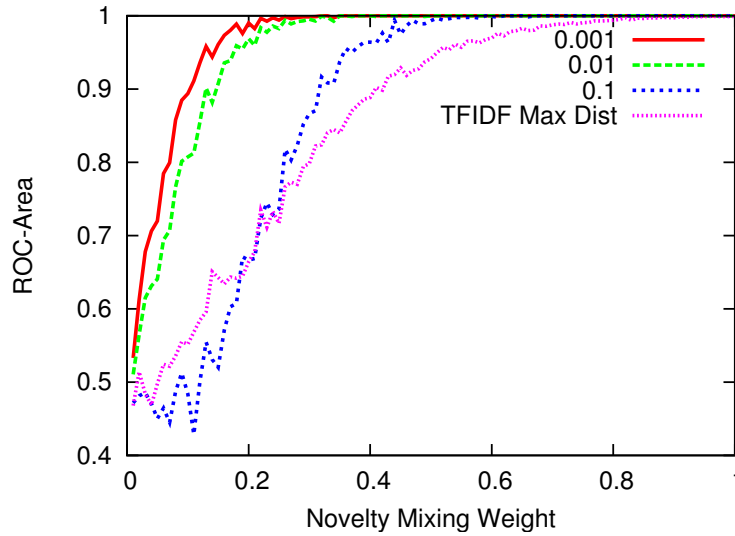


Figure 3.5: ROC-Area analysis of the novelty score for Jelinek-Mercer smoothing with different  $\lambda$  for controlling the amount of corpus smoothing. The baseline is TFIDF Max.

ods paper, then it should be relatively similar to the other kernel methods papers. In fact, it will probably be more similar to these other kernel methods papers than to the background mixture of all 100 previous documents. Therefore, as the amount of novel content increases in a mixture, the mixture vector actually moves closer to this kernel methods cluster, and the TFIDF Min distance decreases, which runs counter to intuition. Such cases occur commonly enough that it causes TFIDF Min to plateau after a certain amount of novelty. On the other hand, if the most dissimilar document were completely orthogonal, with no words in common, then TFIDF Max would be completely useless for scoring novelty, since all the distances would be one. In practice, enough words overlap (even just common words such as “computer” or “experiments”) between the generated documents (and the NIPS vectors) that TFIDF Max works in practice.

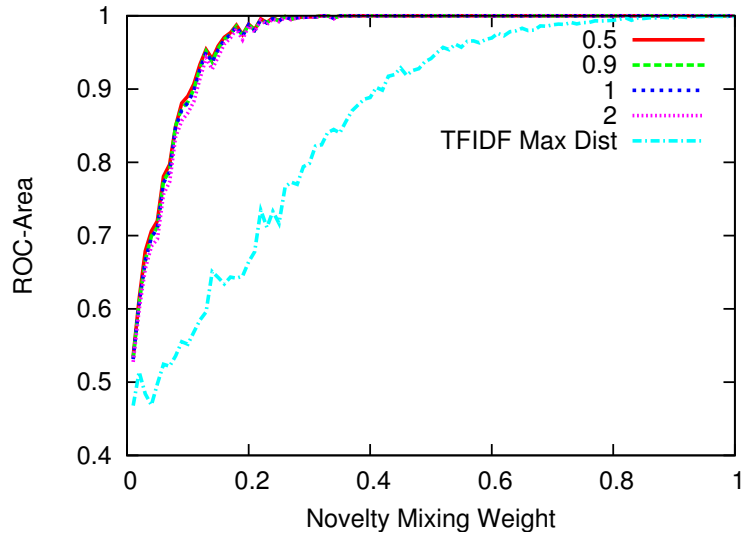


Figure 3.6: ROC-Area analysis of the novelty score for Dirichlet smoothing with different  $\mu$ , compared against TFIDF Maximum Distance.

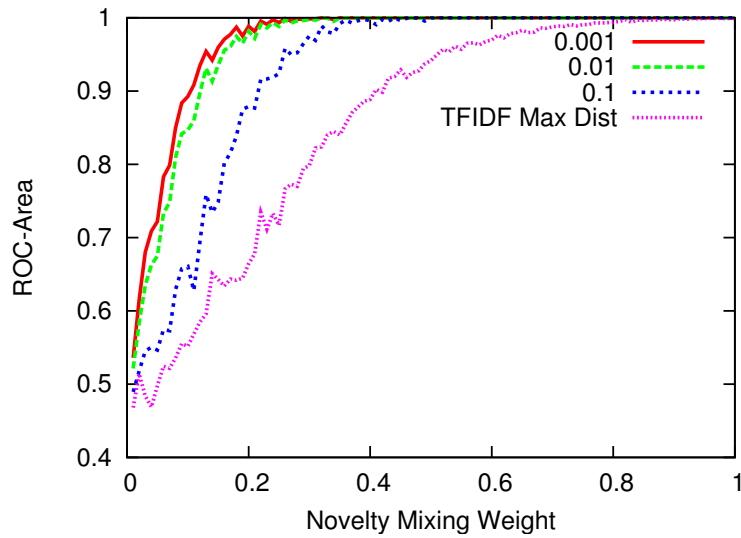


Figure 3.7: ROC-Area analysis of the novelty score for Discount smoothing with different  $\delta$ , compared against TFIDF Maximum Distance.

### Smoothing in Novelty Detection

In novelty detection, since some terms may have estimated probabilities equal to zero, smoothing is quite important. We evaluate the three smoothing methods: Jelinek-

Mercer, Dirichlet, and Discount, to see if there is a clear winner, and what smoothing parameter settings are appropriate for this task.

Figure 3.5 shows that for Jelinek-Mercer smoothing, mixing  $\lambda$  amount of corpus background does quite well for reasonable, small values of  $\lambda$ , such as 0.001 and 0.01. With  $\lambda = 0.1$ , the method does worse than when using smaller values of  $\lambda$ , while being very similar to the TFIDF Max baseline. With large values of  $\lambda$  (not shown), the method becomes worse since the document term probabilities are overwhelmed by the corpus probabilities used to smooth. In practice, one would want a small value for  $\lambda$ .

Figure 3.6 shows that the Dirichlet smoothing method is quite robust for many values of the smoothing parameter  $\mu$ , including the range 0.5 to 2. This range includes Laplace smoothing, where  $\mu = 1$ . Dirichlet smoothing with these various  $\mu$  settings performs similarly to Jelinek-Mercer smoothing with  $\lambda = 0.001$ , as was shown in Figure 3.3. The novelty method with Dirichlet smoothing works much better than the TFIDF Max baseline.

Figure 3.7 shows that for Discount smoothing, as in Jelinek-Mercer smoothing, small values of  $\delta$  such as 0.001 and 0.01 work quite well, and much better than the TFIDF Max baseline. With  $\delta = 0.1$ , the method gets noticeably worse, splitting the difference with the TFIDF Max baseline. For larger values of  $\delta$ , as with Jelinek-Mercer smoothing, the method does worse because of too much smoothing. One would typically choose smaller values for  $\delta$ .

With all three smoothing methods, one must become careful about making the smoothing values too small. If the  $\lambda$ ,  $\mu$ , and  $\delta$  are too small, then there is not enough smoothing, and the probabilities are still practically zero. In these results, the minimum presented values for these parameters are the minimum ones for which the optimization

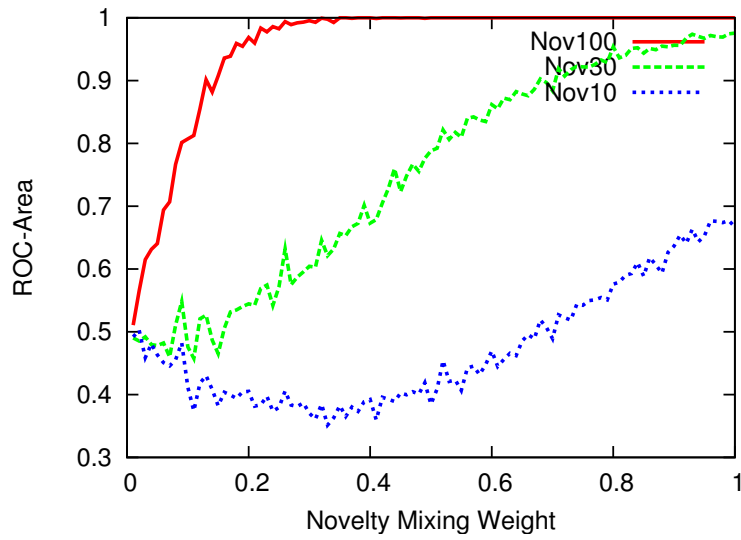


Figure 3.8: ROC-Area analysis of the novelty score for Jelinek-Mercer smoothing with  $\lambda = 0.01$  and 10, 30, and 100 previous documents.

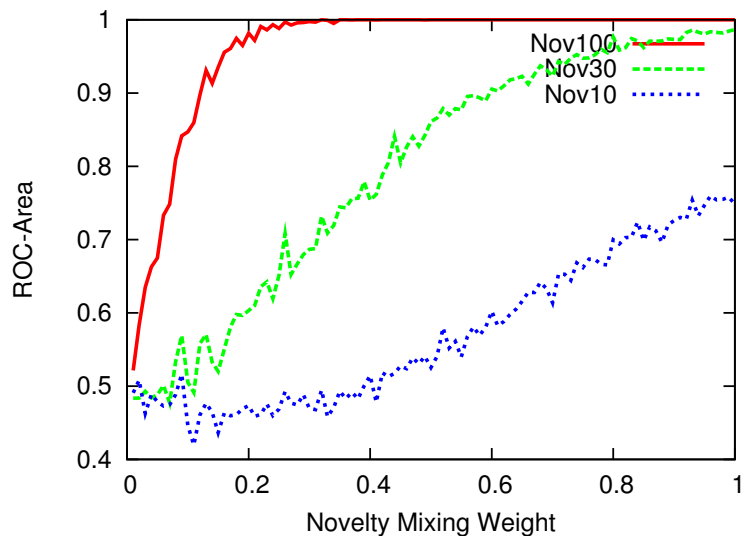


Figure 3.9: ROC-Area analysis of the novelty score for Discount smoothing with  $\delta = 0.01$  and 10, 30, and 100 previous documents.

converged in our experiments on the synthetically-generated data. We also tried smaller values, but then the optimizer ran into trouble because the probabilities were too close to 0.

## Choosing Enough Previous Documents

It is important to choose enough previous documents so that the method has sufficient background information against which to compare for novelty. The earlier experiments used the full set of  $k_p = 100$  previous documents for the analysis. Here, we restrict that set to the most similar 10 or 30 previous documents to determine how well the method can identify novel documents.

Figure 3.8 shows the results for  $k_p \in \{10, 30, 100\}$  previous documents for Jelinek-Mercer smoothing in the novelty method. Obviously, with 100 previous documents, the method does the best. With  $k_p = 30$ , the method does much worse, even somewhat worse than the TFIDF baseline (that used 100 previous documents), but still well above random. With 10 previous documents, the method does much worse, in fact, even much worse than the TFIDF baseline. Setting  $k_p = 10$  gives quite close to random performance.

Figure 3.9 shows the results for  $k_p \in \{10, 30, 100\}$  previous documents with Discount smoothing. The overall results seem quite similar to the case with Jelinek-Mercer smoothing. Using 100 documents is the best and much better than the TFIDF Max baseline. With 30 previous documents, the method is slightly worse than the TFIDF baseline (that used 100 documents). Using 10 documents is much worse than the baseline, and near to, but slightly better than random, especially when the novelty mixing weight is above 0.6 or 0.8.

## Most Novel Documents in NIPS

We present here the list of the most novel documents in NIPS according to the novelty score, using Discount smoothing with  $\delta = 0.01$ . Table 3.2 shows the most novel doc-

Table 3.2: The most novel documents in NIPS according to the KL-Divergence score using Discount smoothing with  $\delta = 0.01$ .

Score $s_n^{(i)}$	Document Title and Author(s)
8.556	“Author Index”
7.448	“Song Learning in Birds” by M. Konishi
7.034	“Author Index”
6.496	“Part VIII Applications”
6.437	“Part I Cognitive Science”
6.353	“A Neural Network to Detect Homologies in Proteins” by Yoshua Bengio, Samy Bengio, Yannick Pouliot, Patrick Agin
6.161	“Author Index”
5.904	“Part II Neuroscience”
5.838	“Author Index”
5.827	“Connectionism for Music and Audition” by Andreas Weigand

uments from NIPS according to the KL-divergence novelty score. Table 3.3 shows the most novel NIPS document from each year. These tables are insightful in seeing what does not work with the novelty method. We had defined the most novel documents as being the ones that differed the most from existing content in preceding documents. These tables show that the most novel documents are often the author indices. Since most authors only publish a small number of times, or even just once, the list of all the author names tends to be highly novel. These documents were included in the 1955 NIPS documents because they were part of the OCR-ed data provided. However, since these are not really proper research publications, we removed author indices, track headings, and introductory table of contents documents from the data, which left 1908 documents.

We ran the KL-Divergence novelty scoring method on this smaller cleaned-up set



Table 3.3: The most novel document per year of NIPS according to the KL-Divergence score using Discount smoothing with  $\delta = 0.01$ .

Document		Score
Year	Document Title and Author(s)	$s_n^{(i)}$
1988	“Author Index”	8.556
1989	“A Neural Network to Detect Homologies in Proteins” by Yoshua Bengio, Samy Bengio, Yannick Pouliot, Patrick Agin	6.353
1990	“Author Index”	5.838
1991	“Author Index”	5.763
1992	“Author Index”	5.247
1993	“Connectionism for Music and Audition” by Andreas Weigand	5.827
1994	“Grammar Learning by a Self-Organizing Network” by Michiro Negishi	5.371
1995	“Author Index”	7.034
1996	“Index of Authors”	5.531
1997	“Part VIII Applications”	6.496

of documents with the same smoothing settings. Tables 3.4 and 3.5 show the highest scoring novel documents overall and per year. Many documents that have high novelty scores indeed seem to be different from the typical content of NIPS. While these documents indeed seem to be novel and different, they are not necessarily the most influential, because these ideas did not necessarily become very popular in following documents. In the next chapter, we will address the combination of novelty with influence to lead to a more refined method for identifying new and interesting ideas.

Table 3.4: The most novel documents in NIPS according to the KL-Divergence score, not including outlier documents, using Discount smoothing with  $\delta = 0.01$ .

Score $s_n^{(i)}$	Document Title and Author(s)
7.459	“Song Learning in Birds” by M. Konishi
6.353	“A Neural Network to Detect Homologies in Proteins” by Yoshua Bengio, Samy Bengio, Yannick Pouliot, Patrick Agin
6.002	“Connectionism for Music and Audition” by Andreas Weigend
5.498	“Neural Architecture” by Valentino Braitenberg
5.358	“Grammar Learning by a Self-Organizing Network” by Michiro Negishi
5.321	“Acoustic-Imaging Computations by Echolocating Bats: Unification of Diversely-Represented Stimulus Features into Whole Images” by James A. Simmons
5.288	“Analytic Solutions to the Formation of Feature-Analysing Cells of a Three-Layer Feedforward Visual Information Processing Neural Net” by D. S. Tang
5.177	“A B-P ANN Commodity Trader” by Joseph E. Collard
5.152	“Harmonet: A Neural Net for Harmonizing Chorales in the Style of J.S. Bach” by Hermann Hild, Johannes Feulner, Wolfram Menzel
5.140	“Stability and Observability” by Max Garzon, Fernanda Botelho

### Decreasing Novelty over Time

As an artifact of our model that measures novelty by term usage patterns, there may be some edge effects for the first few years. The earlier documents seem to have higher novelty scores when compared to later documents. Table 3.6 shows that the yearly average of the novelty score (computed over all documents in the corpus) decreases over time. Additionally, the maximum value decreases, while the minimum value does not obey this trend. In fact, what is probably happening is that the first few years of docu-

Table 3.5: The most novel document per year of NIPS according to the KL-Divergence score, not including outlier documents, using Discount smoothing with  $\delta = 0.01$ .

Document		Score
Year	Document Title and Author(s)	$s_n^{(i)}$
1988	“Song Learning in Birds” by M. Konishi	7.459
1989	“A Neural Network to Detect Homologies in Proteins” by Yoshua Bengio, Samy Bengio, Yannick Pouliot, Patrick Agin	6.353
1990	“A B-P ANN Commodity Trader” by Joseph E. Collard	5.177
1991	“Harmonet: A Neural Net for Harmonizing Chorales in the Style of J.S. Bach” by Hermann Hild, Johannes Feulner, Wolfram Menzel	5.152
1992	“Hidden Markov Models in Molecular Biology: New Algorithms and Applications” by Pierre Baldi, Yves Chauvin, Tim Hunkapiller, Marcella A. McClure	5.013
1993	“Connectionism for Music and Audition” by Andreas Weigend	6.002
1994	“Grammar Learning by a Self-Organizing Network” by Michiro Negishi	5.358
1995	“The Role of Activity in Synaptic Competition at the Neuromuscular Junction” by S. R. H. Joseph, D. J. Willshaw	4.513
1996	“Spectroscopic Detection of Cervical Pre-Cancer through Radial Basis Function Networks” by Kagan Turner, Nirmala Ramanujam, Rebecca Richards-Kortum, Joydeep Ghosh	4.485
1997	“Gradients for Retinotectal Mapping” by Geoffrey J. Goodhill	4.687

ments probably get “novelty credit” for many background terms that were in existence before the first NIPS document was ever written, such as “computer” or “information.”

Table 3.6: Basic statistics for the novelty score for years of NIPS, computed using Discount smoothing with  $\delta = 0.01$ .

Year	Minimum	Maximum	Average
1988	2.124	8.556	3.676
1989	1.405	6.353	3.280
1990	1.663	5.838	3.239
1991	1.834	5.763	3.170
1992	2.011	5.247	3.117
1993	1.808	5.827	3.180
1994	1.745	5.371	2.976
1995	1.949	7.034	2.969
1996	1.540	5.531	2.955
1997	2.116	6.496	3.149
1998	2.165	4.819	2.896
1999	1.855	4.469	2.827
2000	1.942	4.334	2.812

### Novelty in Influential NIPS Documents

In addition, Table 3.7 presents the novelty scores for the most influential NIPS documents. In Table 2.6, we presented a list of the most influential NIPS document per years of the conference. Here, we show their novelty scores and the percentile of their novelty scores as compared against other documents within the same year. Overall, these influential documents typically have percentiles in the middle or the low range of the novelty scores for the year. That suggests that some documents expressed very different and novel ideas, but that the most different ideas typically did not catch on or become very popular. However, the documents that were most important in influencing future

Table 3.7: The KL-Divergence novelty score using Discount smoothing with  $\delta = 0.01$  for the most influential paper per year of NIPS. The percentile is of the novelty score compared against all documents in that year. More details on how these papers are selected were presented in Table 2.6.

Document		Score	
Year	Document Title and Author(s)	$s_n^{(i)}$	Percentile
1988	“Efficient Parallel Learning Algorithms for Neural Networks” by Alan Kramer, A. Sangiovanni-Vincentelli	3.049	18
1989	“Training Stochastic Model Recognition Algorithms as Networks Can Lead to Maximum Mutual Information Estimation of Parameters” by John S. Bridle	2.988	38
1990	“Integrated Modeling and Control Based on Reinforcement Learning” by R. S. Sutton	3.045	36
1991	“Bayesian Model Comparison and Backprop Nets by David J. C. Mackay	3.109	51
1992	“Reinforcement Learning Applied to Linear Quadratic Regulation” by Steven J. Bradtke	3.050	51
1993	“Supervised Learning from Incomplete Data via an EM approach” by Zoubin Ghahramani, Michael I. Jordan	2.977	40
1994	“Reinforcement Learning Algorithm for Partially Observable Markov Decision Problems” by Tommi Jakkola, Sizarad Singhal, Michael I. Jordan	2.534	18
1995	“EM Optimization of Latent-Variable Density Models” by Chris M. Bishop, M. Svensen, Chistopher K.I. Williams	2.545	18
1996	“Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing” by V. Vapnik, Steven E. Golowich, Alex Smola	3.513	87
1997	“EM Algorithms for PCA and SPCA” by Sam Roweis	2.535	16

content in the corpus typically were somewhat different, but not the most different. The one exception is the highly novel Vapnik et al. SVM document from 1996 that was also very influential. Overall, this table is quite telling in terms of just how novel the ideas that became popular were.

### **3.5 Discussion and Future Work**

The novelty-based methods enable people to find documents that have a high concentration of new ideas. As the experiments have shown, the catch is that in practice, novelty itself is not necessarily enough. Since text is intrinsically quite noisy, novelty on the one hand may indeed detect some important original idea, but on the other hand may simply detect some noise or relatively inconsequential ideas in the data. Although finding the most novel documents, while noisy, may be interesting for understanding the breadth of content represented in the corpus, it is not really the right method for focusing attention on the important new ideas. The next chapter will focus on combining novelty with influence in the context of the copy models.

Additionally, with novel terms, we found that TFIDF was comparable or perhaps slightly better than the novelty method based on the Influence Model. TFIDF has the advantage of considering the entire corpus when deciding on which terms are important, while the novelty model can only look at the past and the current document. Because of the long tail of term occurrences, there are many terms that might just occur or not because of chance, making it difficult to sort out rare term occurrence events from truly novel content. Sorting through this noise can be remedied by looking not only at the past, but also into the future as we consider a corpus offline.

### 3.6 Summary

We presented a method based on the Inter-Document Influence Model with Novel Content (Model 2.3) for analyzing documents to figure out what makes them novel. This method can look for novel terms according to the highest-probability terms from the learned novel language model. Another task was to identify how novel each document is. For this task, we proposed a KL-Divergence model for scoring the novelty of each document relative to the best possible explanation of that document's content using existing ideas in the corpus. Overall, these methods seemed to identify content that is different from existing ideas. While these methods were able to identify new and different content, such as the author indices, they did not focus on the most important ideas in the corpus. In the final analysis, these methods were not really appropriate for the task of finding the origins of the ideas that drove the development of the corpus. Consequently, in the following chapter, we will explore methods that leverage both impact and novelty to address this task of analyzing documents to identify original contributions that are both new and important.

## CHAPTER 4

### IDEA ORIGINS IN TEXT

While the methods for novelty detection in document collections seemed plausible, they did not work that well because novelty alone was insufficient to distinguish between novel ideas that had impact and novel ideas that did not influence other documents. For example, in the larger set of documents that include author and subject indices, the author indices often had high novelty scores. Even though this made sense, since many authors publish relatively few papers in NIPS, so that their names were novel, we would rather focus our attention on the content that influenced the overall development of the corpus. This chapter builds on novelty, by combining it with impact, to identify those important ideas that shaped the corpus.

In Ch. 2, we found that the Inter-Document Influence Model (Model 2.2) was able to find influential documents quite well. This chapter refines that model to identify the original contributions that not only influence future documents, but also differ from existing content. Here, an original contribution is defined as combining both novelty and influence. The goal is to help users find the most important ideas in the corpus by pointing out where these ideas originate within document text.

#### **4.1 Introduction**

The key for finding original contributions is to point out each document's novel ideas that ultimately had impact on the future development of the corpus. Anybody can write some spam on a discussion board, which would likely be novel to the discussion (at least the first time), but not particularly interesting. In addition to novelty, measuring the impact of an idea lets us focus on those ideas that are important, or that at least



are interesting to a large number of people. Therefore, our operational definition of an original contribution combines both novelty and impact.

Unlike methods that rely on explicit citations that must be localizable in each document (Mei & Zhai, 2008), our methods require only the text of the documents. This makes them more broadly applicable than citation-based measures (e.g., for email, news). Furthermore, unlike novelty detection methods (Soboroff & Harman, 2003) (e.g., based on TFIDF-style measures), our methods combine novelty with impact, which provides a way of measuring the importance of novel ideas. The originality-detection methods we propose are derived from a probabilistic language model of diachronic corpora – called the Passage Impact Model (PIM), which makes them theoretically well-founded and more extensible than heuristic approaches. The method is evaluated on a corpus of Slashdot discussions, as well as through a blind experiment with human judges on a collection of NIPS research articles. In both experiments, the language modeling approach was found to outperform a heuristic that focuses on novelty detection alone.

## **4.2 Related Work: Summarization and Novelty**

The task of succinctly describing the original contribution of a document relates to several existing research areas, including document summarization, topic detection, topic modeling, and language modeling.

### **4.2.1 Document Summarization**

The largest body of related work is in document summarization (see e.g., (NIST, 2001)). Document summarization methods provide the user with a summary of the entire docu-

ment, including both original and existing ideas, without explicitly making a distinction. The difference between summarization and originality detection is most apparent for documents that do not necessarily contain original content (e.g., textbooks, review articles). While such documents have a summary, their original contribution can be quite different or even non-existent.

### **4.2.2 Novelty Detection**

Another area of related work lies in novelty detection for Topic Detection and Tracking (Allan et al., 1998a; Allan et al., 1998b) in news streams. There, the task is to identify new topics and events as they appear in the news. One major difference is that the Passage Impact Model segments the document to identify a single passage that best describes that document’s original contribution. Thus the inference method can actually find a text description within the document, instead of just marking that the document contains a novel topic. A second difference is that the Passage Impact Model combines novelty with impact, focusing on ideas that not only are novel but also affect the rest of the corpus. The TREC Novelty track (Soboroff & Harman, 2003) solves a different problem, combining novelty and relevance, not novelty and impact.

### **4.2.3 Impact-Based Summaries**

One previous paper has tackled the problem of making “impact-based summaries” (Mei & Zhai, 2008). Their method is based on citation contexts for explicit citations to a document  $d$ . The task is to select the sentence  $s$  in document  $d$  that best describes the contribution of  $d$  that had impact in these citation contexts. That work followed a KL-

divergence-based information retrieval framework where the document  $d$  stands for the corpus, the sentences  $s$  stand for the documents to be retrieved, and the citation context is descriptive of the “query.” The Passage Impact Model is quite different in model and inference, since it does not require citations. Instead, our method is based on an extensible generative and unsupervised language-modeling framework. We start from a generative model of the corpus and derive an inference method to identify the most densely-concentrated original contribution in the document  $d$ . We do not need to use a citation context, as the method is completely text-based.

#### 4.2.4 Topic Modeling

On a higher level, topic models and other language models also provide generative models of corpora. In topic models, however, the focus is on discovering underlying topics, without any explicit notion of originality or impact. Typically, topics are inferred by fitting graphical models with topics as the latent variables. Latent Dirichlet Allocation (LDA) (Blei et al., 2003b; Blei et al., 2003a) and its extensions (Blei & Lafferty, 2005; Blei & Lafferty, 2006) are the most well-known, but there is much other work in topic modeling (Hofmann, 1999; Mann et al., 2006; Wang & McCallum, 2006; Steyvers et al., 2004; Griffiths & Steyvers, 2002; Dietz et al., 2007; McCallum et al., 2005; Mei et al., 2007; Li & McCallum, 2006; Wang et al., 2006; Griffiths et al., 2004). In this sense, topic models describe the relationship between topics and documents, but not the relationships between individual documents. Our Passage Impact Model directly models relationships between documents via a copy process. In this sense it builds on the Inter-Document Influence Model from Ch. 2 and the Citation Model from (Dietz et al., 2007), extending them to recognizing document substructure. We use simple unigram language models in the PIM, but one could also use more complex language models (Manning &

Schuetze, 1999; Hofmann, 1999; Blei et al., 2003b; Jelinek, 1998; Zhai, 2002; Kurland & Lee, 2004; Kurland & Lee, 2006).

### 4.3 Methods

We take a language modeling approach and define a generative model for diachronic corpora. An author writes a new document using a mixture of novel ideas and ideas “copied” from earlier documents. An idea has impact if it is copied (i.e., discussed, elaborated on) by future documents. This picture is one of idea flows, originating in documents with impact and “flowing” to documents based on idea development. We directly model idea flows between documents, without an extra level of the topic as in topic models (Blei et al., 2003b). Identifying the original contribution of a document means separating novel ideas from old ideas, and simultaneously assessing impact. We assume that documents generally contain a key paragraph or sentence(s) that succinctly describe the new idea, and we aim to identify this piece of original text. The following gives more detail on our probabilistic model and inference method.

#### 4.3.1 Passage Impact Model

We propose a generative model of a diachronic corpus that extends the Inter-Document Influence Model with Novel Content 2.3 in Ch. 2 with respect to modeling originality. We model a document  $D^{(i)}$  containing  $n_i$  words as a vector of  $n_i$  random variables  $W^{(i)} = (W_1^{(i)} \cdots W_{n_i}^{(i)})'$ , one per word. Considering the process by which authors write documents, the text can be split into several types: original content that will have impact on following documents, novel content that will not have impact, and content “copied”

from already-existing ideas in the corpus. The location of the original content in  $D^{(i)}$  is denoted by  $Z^{(i)}$ , where  $Z^{(i)} \subseteq \{1 \cdots n_i\}$ . More concretely, the random variables  $W^{(i)}$  are partitioned into two sets:  $Z^{(i)} \subseteq \{1 \cdots n_i\}$  for the indices of the words of  $D^{(i)}$  that are original and have impact, while  $\bar{Z}^{(i)} = \{1 \cdots n_i\} - Z^{(i)}$  contains the rest of  $D^{(i)}$  (i.e., the copied content and the novel content without impact). With these definitions, the document is described by the tuple

$$D^{(i)} = (W^{(i)}, Z^{(i)}) \quad (4.1)$$

and we will now define a probabilistic model of a document  $P(D^{(i)} | D^{(1)} \cdots D^{(i-1)})$ . Each document  $D^{(i)}$  can draw on the ideas already expressed in the existing documents  $D^{(1)} \cdots D^{(i-1)}$  in the corpus. The probability of an entire corpus  $C$  consisting of documents  $D^{(1)} \cdots D^{(n)}$ , can be decomposed as

$$P(C) = \prod_{i=1}^n P(D^{(i)} | D^{(1)} \cdots D^{(i-1)}). \quad (4.2)$$

We decompose the probability for a single document  $D^{(i)}$  into

$$\begin{aligned} P(D^{(i)} | D^{(1)} \cdots D^{(i-1)}) &= P(W^{(i)}, Z^{(i)} | D^{(1)} \cdots D^{(i-1)}) \\ &= P(W^{(i)} | Z^{(i)}, D^{(1)} \cdots D^{(i-1)}) P(Z^{(i)}) \end{aligned}$$

since the document text  $W^{(i)}$  depends on the previous documents, but the author's selection of placement of original content is independent of previous documents. Prior information about the placement of  $Z^{(i)}$  in the document can be encoded in  $P(Z^{(i)})$ . Furthermore, in the inference described below, the quantity  $P(Z^{(i)})$  can be used to encode constraints on the form of original content summary that is desirable (e.g., a single sentence or a single paragraph).

Words in the original portion  $Z^{(i)}$  are generated from a unigram language model with word probabilities  $\tilde{\theta}^{(i)}$ . The rest of the document (i.e. the words indexed by  $\bar{Z}^{(i)}$ ) comes from a mixture of existing ideas and text that is novel but without impact. That is, the

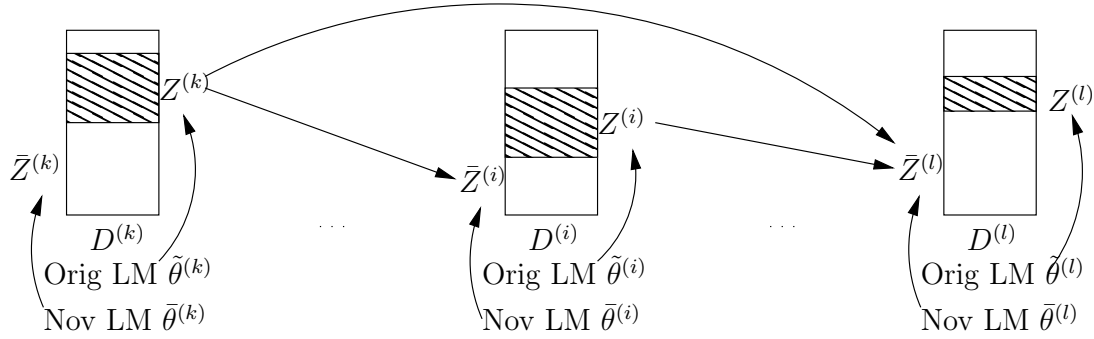


Figure 4.1: The generative process for a corpus. Document  $d^{(i)}$  is the current document, while  $d^{(k)}$  precede  $d^{(i)}$  in time and  $d^{(l)}$  follow  $d^{(i)}$ . The shaded boxes are original content  $Z^{(i)}$ , while the rest of the documents form  $\bar{Z}^{(i)}$ . The arrows depict the copy process.

words indexed by  $\bar{Z}^{(i)}$  are drawn from a mixture of a novel unigram model  $\bar{\theta}^{(i)}$  (new but without impact) and words copied from the original sections of prior documents. Words are drawn uniformly and independently in this copy process so that it can also be described by a unigram model with parameters  $\hat{\theta}^{(k)}$  for each prior document  $D^{(k)}$ . The document-specific mixing weights  $\pi^{(i)}$  are  $(\pi_n^{(i)}, \pi_k^{(i)})$  for  $\bar{\theta}^{(i)}$  and  $\hat{\theta}^{(k)}$ , respectively.

With the assumption that text is generated from these unigram multinomial language models, the generative model of the text given  $Z^{(i)}$  and the existing corpus at time  $i$  is

$$P(W^{(i)} | Z^{(i)}, D^{(1)} \dots D^{(i-1)}) = \prod_{j \in Z^{(i)}} \binom{\tilde{\theta}_j^{(i)}}{w_j^{(i)}} \prod_{j \in \bar{Z}^{(i)}} \left( \pi_n^{(i)} \bar{\theta}_j^{(i)} + \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_j^{(k)} \right).$$

Figure 4.1 illustrates the generative process at document  $d^{(i)}$ , showing how  $d^{(i)}$  copies content from the original part  $Z^{(k)}$  of earlier documents  $d^{(k)}$  and showing how terms indexed by  $Z^{(i)}$  are copied by later documents  $d^{(l)}$ . We summarize this generative process of a diachronic corpus in the Passage Impact Model.

**Model 4.1 (PASSAGE IMPACT MODEL)**

A corpus  $C = (D^{(1)} \dots D^{(n)})$  of temporally-sorted documents  $D^{(i)} = (W^{(i)}, Z^{(i)})$ , each having parameters  $(\tilde{\theta}^{(i)}, \bar{\theta}^{(i)}, \pi^{(i)})$ , has probability  $P(C) = \prod_{i=1}^n P(D^{(i)} | D^{(1)} \dots D^{(i-1)})$

where

$$P(D^{(i)} | D^{(1)} \dots D^{(i-1)}) = \prod_{j \in z^{(i)}} \left( \tilde{\theta}_{w_j^{(i)}}^{(i)} \right) \prod_{j \in \bar{z}^{(i)}} \left( \pi_n^{(i)} \bar{\theta}_{w_j^{(i)}}^{(i)} + \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_{w_j^{(i)}}^{(k)} \right) P(Z^{(i)})$$

and where  $\hat{\theta}_w^{(k)}$  is the probability of uniformly drawing word  $w$  from the words in the original section  $z^{(k)}$  of document  $D^{(k)}$ . Note that  $\pi_n^{(i)} + \sum_k \pi_k^{(i)} = 1$ ,  $\sum_j \tilde{\theta}_j^{(i)} = 1$ , and  $\sum_j \bar{\theta}_j^{(i)} = 1$ .

### 4.3.2 Inference

Using the Passage Impact Model, we are primarily interested in inferring the subset  $Z^{(i)}$  of words in  $D^{(i)}$  where the original contribution is most succinctly contained. The only observed quantity is the text  $w^{(1)} \dots w^{(n)}$  of all documents. We use maximum-likelihood inference based on Model 4.1 for inferring  $Z^{(1)} \dots Z^{(n)}$  by maximizing  $P(D^{(1)} \dots D^{(n)})$  given  $w^{(1)} \dots w^{(n)}$  w.r.t.  $Z^{(i)}$ ,  $\tilde{\theta}^{(i)}$ ,  $\bar{\theta}^{(i)}$ , and  $\pi^{(i)}$ . Applying Bayes rule and independence assumptions involving the placement of original content  $Z^{(i)}$  in different documents  $D^{(1)} \dots D^{(n)}$ , the inferred original content  $Z^{(i)*}$  is given by the following:

$$\begin{aligned} (Z^{(1)*} \dots Z^{(n)*}) &= \operatorname{argmax}_{Z^{(1)} \dots Z^{(n)}} \sup_{(\tilde{\theta}, \bar{\theta}, \pi)} P(w^{(1)} \dots w^{(n)} | Z^{(1)} \dots Z^{(n)}) P(Z^{(1)} \dots Z^{(n)}) \\ &= \operatorname{argmax}_{Z^{(1)} \dots Z^{(n)}} \sup_{(\tilde{\theta}, \bar{\theta}, \pi)} P(w^{(1)} \dots w^{(n)} | Z^{(1)} \dots Z^{(n)}) P(Z^{(1)}) \dots P(Z^{(n)}) \end{aligned}$$

Note that we do not explicitly include the parameters  $\tilde{\theta}$ ,  $\bar{\theta}$ , and  $\pi$  in the notation for improved readability, since their dependence is straightforward. To avoid the intractable simultaneous maximization over all  $(Z^{(1)} \dots Z^{(n)})$ , we introduce some simplifying assumptions that allow independent optimization for each  $Z^{(i)}$ . First, we assume that for all prior documents  $d^{(1)} \dots d^{(i-1)}$ , the copy probabilities  $\hat{\theta}^{(1)} \dots \hat{\theta}^{(i-1)}$  can be approximately

estimated from the full set of words  $w^{(1)} \dots w^{(i-1)}$ , respectively, not merely the words indexed by the original markers  $z^{(1)} \dots z^{(i-1)}$ . In practice, this assumption can be expected to have only minor impact<sup>1</sup>, and it can be removed if  $z^{(1)} \dots z^{(i-1)}$  are already known.

With this assumption, we have that for any  $i$

$$\begin{aligned} (Z^{(i)*} \dots Z^{(n)*}) &= \operatorname{argmax}_{Z^{(i)} \dots Z^{(n)}} \sup_{Z^{(1)} \dots Z^{(i-1)}} \sup_{(\bar{\theta}, \bar{\theta}, \pi)} P(w^{(1)} \dots w^{(n)} | Z^{(1)} \dots Z^{(n)}) P(Z^{(1)}) \dots P(Z^{(n)}) \\ &= \operatorname{argmax}_{Z^{(i)} \dots Z^{(n)}} \sup_{(\bar{\theta}, \bar{\theta}, \pi)} P(w^{(i)} \dots w^{(n)} | Z^{(i)} \dots Z^{(n)}, \hat{\theta}^{(1)} \dots \hat{\theta}^{(i-1)}) P(Z^{(i)}) \dots P(Z^{(n)}) \end{aligned}$$

Second, we introduce a simplified model for the future documents  $D^{(i+1)} \dots D^{(n)}$  so that one can maximize over  $Z^{(i)}$  independently. When inferring  $Z^{(i)}$ , modeling exactly how future documents  $D^{(l)}$ ,  $l > i$ , had impact on each other is of minor importance, so that we do not model their  $Z^{(l)}$ . Instead, we assume that the original and novel content of future documents comes from a multinomial mixture, which can be captured by a single multinomial language model  $\bar{\theta}^{(l)}$ . Thus, each  $D^{(l)}$  depends only on the documents  $D^{(1)} \dots D^{(i)}$ , and

$$P(w^{(i+1)} \dots w^{(n)} | Z^{(i)} \dots Z^{(n)}, \hat{\theta}^{(1)} \dots \hat{\theta}^{(i-1)}) = \prod_{l=i+1}^n P(w^{(l)} | Z^{(i)}, w^{(i)}, \hat{\theta}^{(1)} \dots \hat{\theta}^{(i-1)})$$

Putting all of these assumptions together, we can rewrite the objective function as the likelihood of the documents in the corpus starting from  $D^{(i)}$ , given all the documents that precede  $D^{(i)}$ , which is  $P(D^{(i)} \dots D^{(n)} | D^{(1)} \dots D^{(i-1)})$ . We express this likelihood using

---

<sup>1</sup>Since each document can be a mixture of original content and previous content, when estimating  $\hat{\theta}^{(i)}$  from the entire document, it is equal to the true  $\hat{\theta}^{(i)}$ , mixed with some previous content that would have come from the  $\hat{\theta}^{(i)}$  of even earlier documents in the corpus. This assumption means that the  $\hat{\theta}^{(i)}$  also could include some content from  $\bar{\theta}^{(i)}$ . However, if this portion's mixture component is relatively small, the  $\hat{\theta}^{(i)}$  will still be quite faithful to the Passage Impact Model's definition.



the parameters  $(\tilde{\theta}^{(i)}, \bar{\theta}^{(i)}, \pi^{(i)})$  as follows:

$$\begin{aligned}
Z^{(i)*} &= \operatorname{argmax}_{Z^{(i)}} \sup_{(\tilde{\theta}, \bar{\theta}, \pi)} P(Z^{(i)}) P(w^{(i)} | Z^{(i)}, \hat{\theta}^{(1)} \dots \hat{\theta}^{(i-1)}) \prod_{l=i+1}^n P(w^{(l)} | Z^{(i)}, w^{(i)}, \hat{\theta}^{(1)} \dots \hat{\theta}^{(i-1)}) \\
&= \operatorname{argmax}_{Z^{(i)}} \sup_{(\tilde{\theta}, \bar{\theta}, \pi)} \left[ P(Z^{(i)}) \prod_{j \in z^{(i)}} \left( \tilde{\theta}_{w_j^{(i)}}^{(i)} \right) \prod_{j \in \bar{z}^{(i)}} \left( \pi_n^{(i)} \bar{\theta}_{w_j^{(i)}}^{(i)} + \sum_{k=1}^{i-1} \pi_k^{(i)} \hat{\theta}_{w_j^{(i)}}^{(k)} \right) \right. \\
&\quad \left. \prod_{l=i+1}^n \prod_{j=1}^{n_l} \left( \pi_n^{(l)} \bar{\theta}_{w_j^{(l)}}^{(l)} + \sum_{k=1}^i \pi_k^{(l)} \hat{\theta}_{w_j^{(l)}}^{(k)} \right) \right] \tag{4.3}
\end{aligned}$$

Note that the various  $\pi^{(\cdot)}$  and  $\bar{\theta}^{(\cdot)}$ , as well as  $\tilde{\theta}^{(i)}$ , are linearly constrained to form proper probability distributions, and that  $\hat{\theta}^{(i)}$  can be computed in closed form for a given  $z^{(i)}$ . For a fixed  $z^{(i)}$ , the above optimization problem is convex and has no local optima. The prior  $P(Z^{(i)})$  can be used to enforce a particular form of original content description (e.g., that the algorithm has to select a whole paragraph or a single sentence).

### 4.3.3 Implementation Details

When solving the optimization problem, the method can efficiently find the maximum likelihood if given a specific  $z^{(i)}$ . In the following, we therefore give non-zero prior  $P(Z^{(i)})$  only to a fairly small number of  $z^{(i)}$  that can be enumerated explicitly. This allows us to find the globally optimal solution of Eq. 4.3. In particular, we break documents into consecutive passages of equal length, which we denote  $s^1 \dots s^K$ . We set  $P(Z^{(i)} = s^k)$  to be uniform for each  $k = 1 \dots K$ , with all other  $P(z^{(i)}) = 0$ . One could also define a non-uniform prior over the candidate passages  $z^{(i)}$  to encode additional knowledge (e.g., bias toward the beginning or end of the document). With this particular assumption on  $z^{(i)}$ , the entire likelihood maximization can now be reduced to a sequence of convex problems, one per  $s^k$ . The solution to this sequence of optimizations is the global maximum likelihood across the passages. We use the general software optimization tool

MOSEK to solve these convex optimizations (MOSEK, 2008).

While the individual problems are convex, for efficiency reasons, we have to consider the number of parameters in the Passage Impact Model. Therefore, when performing inference on document  $d^{(i)}$ , instead of using the full set of previous documents  $\{d^{(1)} \dots d^{(i-1)}\}$ , we choose the set of  $k_p$  nearest neighbors from these documents according to cosine similarity. The document indices for these  $k_p$  documents are given in the set  $\mathcal{P}$ . Besides  $d^{(i)}$ , the optimization also uses the likelihood of generating the documents  $d^{(i+1)} \dots d^{(n)}$ . Each of these “future” documents  $d^{(l)}$  has its own set of mixing weights and set of previous documents, again chosen from the documents  $\{d^{(1)} \dots d^{(i-1)}\}$  nearest to  $d^{(l)}$  by cosine similarity. While we do not use the following strategies for improving efficiency, one could further reduce the size of the optimization problem. For example, it is possible to consider a Passage Aggregated Impact Model, wherein all future text is “lumped” together into one single “document” for inference. Then, there would only be a single set of future document parameters. Equivalently, we could constrain all future documents to have the same mixing weights and choose the set of previous neighbors as those most similar to the concatenation of all future documents. There is a tradeoff between using more information in more future documents vs. using more parameters for a specific set of interesting previous documents.

## 4.4 Experiments

We conducted experiments to test the Passage Impact Model on both synthetic and real data from research publications and news articles.

### 4.4.1 Analyzing the Model with Synthetic Data

We use synthetic data to explore the range of problems and parameters under which the methods work effectively and robustly. The synthetic data is generated with underlying language models from documents in the full-text proceedings of the Neural Information Processing Systems (NIPS) conference (NIPS Online, 2000) between 1987-2000. NIPS has 1955 documents with text obtained by OCR, resulting in 74731 unique words (multi-character alphabetic strings), except without stopwords.

To generate a document  $d^{(i)}$ , we selected a NIPS document  $d$  randomly and set the original language model  $\tilde{\theta}^{(i)}$  for  $d^{(i)}$  to be the distribution of words in  $d$ . The words indexed in  $Z^{(i)}$  are then generated according to  $\tilde{\theta}^{(i)}$ . For  $\bar{Z}^{(i)}$ , we set the novel language models  $\bar{\theta}^{(i)}$  and each  $\bar{\theta}^{(l)}$  similarly, with each document selected for  $\bar{\theta}^{(l)}$  following NIPS document  $d$  in time. The mixing weights  $\pi_k^{(i)}$  are selected uniformly at random, except for explicitly exploring  $\pi_i^{(l)}$ ,  $l > i$ , (how much future documents  $d^{(l)}$  copy from  $d^{(i)}$ ) and  $\pi_n^{(i)}$  (how much novel but not original content  $d^{(i)}$  has) according to the values they might take in practice.

The structure of  $Z^{(i)}$  and  $\bar{Z}^{(i)}$  takes the form of  $K = 20$  passages with  $L$  words per passage. In the simplest case,  $Z^{(i)}$  marks exactly one passage as original. In addition, we test scenarios where the original content is more diffused through the document, which poses a challenge in inference. One crucial assumption of our method is that the prior  $P(Z^{(i)})$  used during inference matches the data-generating process. However, the inference procedure as implemented above aims to find a single passage containing all the original content, while the true  $Z^{(i)}$  might diffuse it over other passages. To test the robustness of inference w.r.t. the degree of diffusion, we include a fraction  $\delta$  of original content in the (mostly) non-original passages in data generation, but not during inference.

Evaluation on the synthetic data uses the percentage of (mostly) non-original passages with a greater likelihood than the original passage likelihood. Random performance would be that half of the non-original passages are misranked, resulting in a score of 50%. The error values show one standard error.

### **Impact Is Critical**

In the first experiment, we explore the difference between pure novelty detection vs. the additional use of impact when identifying  $Z^{(i)}$ . When not using any future documents, our method might still be able to identify  $Z^{(i)}$  merely by fitting the mixture model and detecting that  $Z^{(i)}$  cannot be expressed as a mixture of previous documents. In this setting, our method becomes a pure novelty detection method. However, Table 4.1 shows that the signal from novelty alone is much weaker than novelty combined with impact. While the performance is better than random when no future documents are used ( $k_F = 0$ ), detection accuracy substantially improves when future documents and impact are considered by the method. The table shows that two future documents that copy 5% of their content from  $d^{(i)}$  already provide a robust signal.

### **More Information in Longer Passages**

We would like to determine the size of the original passage for which the Passage Impact Model can perform well. Users may be interested in descriptions anywhere from one or more sentences to paragraphs. Table 4.2 shows that, in general, when performing inference on longer passages, the method is able to perform more accurately. The method performs very well for passages as short as 50 words. However, for very short passages of length 25 words, there is some drop in accuracy. Longer passages – and therefore

Table 4.1: Percentage of misranked non-original passages. Passage length  $L = 100$ ,  $\delta = 0.2$ ,  $\pi_n^{(i)} = 0.5$ ,  $\pi_i^{(l)} = 0.05$ , and  $\pi_n^{(l)} = 0.6$ . 10 future documents  $d^{(l)}$  were generated, and inference used the  $k_F$  documents  $d^{(l)}$  most (cosine) similar to  $d^{(i)}$ .

$k_F$	% Err $\pm$ One Std Err
0	37.89 $\pm$ 3.23
1	2.95 $\pm$ 0.78
2	0.26 $\pm$ 0.16
5	0.00 $\pm$ 0.00
10	0.16 $\pm$ 0.16

Table 4.2: Percentage of misranked non-original passages with  $k_F = 2$  future documents. The data was generated with  $\delta = 0.2$ ,  $\pi_n^{(i)} = 0.5$ ,  $\pi_i^{(l)} = 0.05$ , and  $\pi_n^{(l)} = 0.6$ .

Length	% Err $\pm$ One Std Err
25	8.16 $\pm$ 1.61
50	2.26 $\pm$ 0.65
100	1.00 $\pm$ 0.99
400	2.26 $\pm$ 0.74

longer documents – provide more observations, and it is less likely that the method will overfit to a few random draws.

### **Diffusiveness of Original Content in $d^{(i)}$**

The inference method searches for a single passage that contains the original contribution, but realistic documents will have original content spread throughout all passages.

Table 4.3: Percentage of misranked non-original passages.  $k_F = 2$  future documents, passages length  $L = 100$  words,  $\pi_n^{(i)} = 0.5$ ,  $\pi_i^{(l)} = 0.05$ , and  $\pi_n^{(l)} = 0.6$ .

$\delta$	% Err $\pm$ One Std Err
0.1	0.00 $\pm$ 0.00
0.2	0.00 $\pm$ 0.00
0.3	4.74 $\pm$ 1.21
0.4	24.89 $\pm$ 2.80
0.5	45.26 $\pm$ 3.38

How much original content in other passages can our inference method tolerate? Table 4.3 shows that the method is very robust towards small to moderate diffusion. Even as  $\delta$  increases to 0.3 (i.e., 30% of each of the other passages is original content), the method is still quite accurate. After that, performance degrades rather quickly, at least when only two future documents are used.

### How Much Copying Is Necessary?

As shown above, the Passage Impact Model relies on future documents copying the ideas expressed in the original contribution of  $d^{(i)}$ . How much must each future document copy to provide a sufficient signal? Table 4.4 shows that the method performs with minimal errors for many values of  $\pi_i^{(l)}$ , even in the situation where future documents copy only 5% of their content (i.e., 100 words) from  $d^{(i)}$ . At lower values for copying, the percentage of correctly ranked passages smoothly decreases. As  $\pi_i^{(l)}$  approaches 0, the method becomes essentially equivalent to a novelty detection method that does not using any future documents.

Table 4.4: Percentage of misranked non-original passages.  $k_F = 2$  future documents, passage length  $L = 100$  words,  $\delta = 0.2$ ,  $\pi_n^{(i)} = 0.5$ , and  $\pi_n^{(l)} = 0.6$ .

$\pi_i^{(l)}$	% Err $\pm$ One Std Err
0.005	34.37 $\pm$ 3.16
0.01	28.58 $\pm$ 2.97
0.02	9.16 $\pm$ 1.41
0.05	0.11 $\pm$ 0.07
0.1	0.00 $\pm$ 0.00
0.2	0.00 $\pm$ 0.00

#### 4.4.2 Predicting Quotations in Slashdot Discussions

Besides synthetic data, we also evaluate on the real world dataset of news articles linked to on Slashdot under the Games topic. When users post an entry, they often link to some article on the Web, and sometimes quote directly from it. Then other users read and respond to these postings in a discussion board format. We collect linked-to web documents and discussions from the Games topic where the original poster directly quotes from a linked-to document. We regard the sentences in the human-selected direct quotations as the label for the original content  $z^{(i)}$  of the web document  $d^{(i)}$ .

We collected a set of 61 documents from the Games topic of Slashdot. These are the entries posted from August 2008 through February 2009, inclusive, where the initial entry quotes a portion of the referenced article. The documents are the referenced articles. In addition, we collect the first page of the user discussion on this topic, as selected by Slashdot. Figure 4.2 shows a screenshot of Slashdot that depicts the data we collected.


An original post including a quotation	Part of the discussion
<p><b>Simulating Emotions Within Games</b></p> <p>Posted by Soulskill on Thu Jan 29, 2009 08:06 AM from the dreams-of-electric-sheep dept.</p> <p>Gamasutra is running an opinion piece about <a href="#">the way video games handle simulated emotions</a>. Most often, an non-player character's emotional state is used to either tell a story or to drive gameplay. The author suggests that as both concepts become more complex in modern games, the simulation of emotions must also become more dynamic to remain interesting. Quoting:</p> <p>"Most of our emotional simulations use a simple sensation/calculation/behavior loop. Someone says or does something to a character; this influences his emotional state; he acts upon his feelings. His emotional state then reverts to a more neutral state over time (I was angry half an hour ago, but I've calmed down now), or changes again in response to another sensation. If these systems are really simple they produce absurd results: a character is furious one moment and cheerful a second later, like a Warner Brothers cartoon character. This is the kind of thing you get with finite state machines. This approach doesn't take into account the fact that behavior itself changes emotions. Behavior is not merely an output to be exhibited; it also affects how we feel. It feeds back into our emotional state."</p> 	<ol style="list-style-type: none"> <li>1. Finite state machines will be unrealistically simple when simulating emotional responses.</li> <li>2. Behavioural-feedback is a necessary condition for realistic emotional displays.</li> </ol> <p>Point number 1 is unwarranted. Finite state machines may elaborate their input at an arbitrarily <i>-finite</i> may still be <i>very large</i>. Part of such an elaboration, of course, may be inner transitions be amount to behavioural-feedback. There is nothing intrinsically <i>un-dynamic</i> to FSM.</p> <p><a href="#">7 hidden comments</a></p> <p><b>Re: Don't hurt the feelings of FSMs (Score: 4, Interesting)</b> by <a href="#">Yvanhoe (564877)</a> on Thursday January 29, @ 10:10AM (#26652669) <a href="#">journal</a></p> <p>Dwarf Fortress uses ASCII characters to display the actors and their various states. It is. IT is not about the graphical feedback, it is about the behavior: once you see some throwing everything around, you know that something is wrong with him. When you see sleeping side by side in the same room, you suspect that something is going on. It is behaviors.</p>

Figure 4.2: Left: A post that quotes from article  $d^{(i)}$  by the link “the way video games handle simulated emotions.” The label for the original content  $z^{(i)}$  in  $d^{(i)}$  is the quotation text. Right: Part of the discussion to be used as the future document  $d^{(l)}$ .

## Experiment Setup

To do inference on Slashdot data, we sort the fulltext, linked-to news articles by their posting date. For each article, we use the Passage Impact Method to rank all the sentences in the linked-to web content  $d^{(i)}$  by their likelihood under the model. The previous documents  $d^{(1)} \dots d^{(i-1)}$  in this setting are the web content that have been linked to in earlier discussions. The future content  $d^{(i+1)}$  in this experiment is the user discussion on this posting, except that any direct quotations from the fulltext article have been removed. The user discussion may not contain all the comments, but only those that have been voted up enough to be selected to appear with the posting. We collected seven months (August 2008 to February 2009, inclusive) of articles that satisfy these criteria from the Games subtopic of Slashdot, which netted a corpus of 61 web documents with their associated discussions.



## Evaluation Method

For evaluation, we rank the sentences in the fulltext article in decreasing order of likelihood. The user quotations typically contain no more than a handful of sentences, but often more than one. Thus, this implementation differs from the model where we assume that there is a single original contribution marked in the passage  $Z^{(i)}$ . As a baseline, we compare against a simple heuristic that identifies novelty. In particular, we rank the sentences by a TFIDF score given by the sum of each sentence term's IDF value. Then, since we have the labels of the true original sentences, we evaluate using the standard metrics of precision and recall at certain points in the ranking. Precision at a point in a ranking is defined to be the number of original sentences at that position in the ranking divided by the total number of sentences up to that point. For a point near the top of the ranking, precision measures whether the sentences that the method most confidently predicts as original are indeed original. Thus we report results for Prec@2. Recall at a point in the ranking is defined to be the number of original sentences at that position in the ranking divided by the total number of original sentences in the document. Recall measures how well the method can find all the original content in the document. Since each labeled quotation typically contains several sentences, we report results for Rec@10.

## Results

The Prec@2 results in Table 4.5 show that the Passage Impact Model outperforms the TFIDF heuristic baseline for predicting the human-selected sentences at the very top of the ranking. For the task of finding a description consisting of a few good sentences that succinctly describe the original content of a news article, the Passage Impact Model is better than the baseline. The PIM also significantly outperforms the baseline when

Table 4.5: Prec@2 and Rec@10 are based on the predicted ranking of sentences by likelihood and TFIDF sum. Original sentences are the ones quoted word-for-word from the article. Results are for  $\pi_n^{(i)} = 0.2$  and  $\pi_n^{(l)} = 0.001$ .

	Prec@2 $\pm$ One Std Err	Rec@10 $\pm$ One Std Err
PIM	22.13 $\pm$ 3.38	36.09 $\pm$ 3.61
TFIDF	9.84 $\pm$ 3.03	25.01 $\pm$ 4.04
RAND	10.63 $\pm$ 1.10	23.92 $\pm$ 2.27

Table 4.6: Comparing the PIM with future documents, and PIM as a novelty detection method (without future documents). Results are for  $\pi_n^{(i)} = 0.2$  and  $\pi_n^{(l)} = 0.001$ .

	Prec@2 $\pm$ One Std Err	Rec@10 $\pm$ One Std Err
PIM Impact	22.13 $\pm$ 3.38	36.09 $\pm$ 3.61
PIM Novelty	9.84 $\pm$ 3.03	28.04 $\pm$ 4.24

trying to find most of the original content, as measured by Rec@10.

### Importance of Impact Component

Similar to the experiment with synthetic data, the use of impact substantially improves the performance over pure novelty detection. Table 4.6 compares the results when using the discussion for detecting impact with the results when no future documents are used. Using the discussion significantly improves the precision of the method.

Table 4.7: Prec@2 and Rec@10 for various amounts of assumed novel content  $\pi_n^{(i)}$  in  $d^{(i)}$ . Sentences are marked as original if they appear word-for-word as in the linked article. Results are for  $\pi_n^{(l)} = 0.001$ .

$\pi_n^{(i)}$	Prec@2 $\pm$ One Std Err	Rec@10 $\pm$ One Std Err
0.01	18.85 $\pm$ 3.10	35.51 $\pm$ 3.55
0.05	20.49 $\pm$ 3.15	36.03 $\pm$ 3.57
0.2	22.13 $\pm$ 3.38	36.09 $\pm$ 3.61
0.8	22.95 $\pm$ 3.39	36.45 $\pm$ 3.63
0.9	22.95 $\pm$ 3.39	36.45 $\pm$ 3.63

### Robustness with respect to amount of novel content in $d^{(i)}$

During inference, the method needs to assume a mixture weight for the novel content in the non-original text  $\bar{Z}^{(i)}$ . How sensitive is the method to the selection of this parameter? Table 4.7 shows that the method is robust and provides good results for a wide range of values for  $\pi_n^{(i)}$ .

### Minor Effect of Novel Language Model in Future Documents

Similarly, since Slashdot discussions are somewhat notorious for getting off topic at times, we evaluated whether changing the amount of novel content in the “future document,” i.e., the discussion makes a difference. As it turns out, Table 4.8 shows that for a wide range of novel content mixing weights  $\pi_n^{(l)}$ , the method is quite robust. The model is able to focus on the portions that the discussion derives from the underlying linked article.

Table 4.8: Prec@2 and Rec@10 for various mixing weights  $\pi_n^{(l)}$  for the noise model in fitting future documents. Sentences are marked as original if they appear word-for-word as in the linked article. The results are reported for  $\pi_n^{(i)} = 0.2$ .

$\pi_n^{(l)}$	Prec@2 $\pm$ One Std Err	Rec@10 $\pm$ One Std Err
0.0001	20.49 $\pm$ 3.15	36.77 $\pm$ 3.58
0.001	22.13 $\pm$ 3.38	36.09 $\pm$ 3.61
0.01	16.39 $\pm$ 3.42	34.55 $\pm$ 3.73
0.1	18.03 $\pm$ 3.29	30.34 $\pm$ 3.47
0.5	20.49 $\pm$ 3.73	31.04 $\pm$ 3.47

### 4.4.3 A User Study

While the Slashdot data provided a reasonable mechanism for inferring ground-truth labels, the most direct evaluation is by explicit human judgment. Therefore, we conducted an experiment with human judges to evaluate the Passage Impact Model on a corpus containing all 1955 papers from the NIPS conference (NIPS Online, 2000) between 1987-2000. In a blind experiment, we asked judges to compare passages extracted by the PIM to those extracted by the TFIDF heuristic regarding how well they summarize the original contribution of a NIPS paper.

#### Experiment Setup

Since breaking documents into paragraphs is non-trivial, especially when they are OCR-ed and have many math equations, we arbitrarily defined passages as consecutive blocks of text of length  $L = 100$  (non-stopword) words. On average, there are 14 passages per document.

For inference using the Passage Impact Model, we constrained the novel  $\bar{\theta}^{(i)}$  and original  $\tilde{\theta}^{(i)}$  language models to be equal because research publications typically discuss original contributions at length. Ideally, the identified passage should list the paper’s contributions or conclusions. (Although the abstract has original content, it mostly focuses on placing the paper with the context of existing ideas.) The future document novelty mixing weight of  $\pi_n^{(l)} = 0.01$  is small to force the model to “explain” the content of future documents  $d^{(l)}$  by identifying copied ideas. For efficiency, we used  $k_F = 5$  future documents. We compare against the TFIDF heuristic baseline. Each paper’s passages predicted by the PIM and the baseline were highlighted, and three judges selected which passage better summarized the paper’s original contribution. The annotators are machine learning graduate students familiar with the corpus and do not include anyone involved in this project.

Since the judgment process is time-consuming, we selected a subset of NIPS publications for evaluation. We ranked all NIPS publications by their number of intra-corpus citations and selected the top 50 most-cited documents. The first publication is “Optimal Brain Damage” by Le Cun, Denker, and Solla, with 27 citations. The entire set of 50 documents includes documents down to those with only 5 intra-1987-to-2000 NIPS citations. The PIM and the baseline selected the same passage on two documents, so we use the remaining 48 for evaluation.

## Results

On these 48 documents, the human judges preferred the Passage Impact Method over the baseline 58.33% of the time, with one standard error of 3.54%. Thus the judges significantly prefer the PIM over the baseline. To analyze the results more closely, we separated the 48 evaluation documents into two sets. On 20 documents, all three

annotators (independently) agreed on a single passage. For these, they preferred the PIM 70% of the time. On the other 28 documents, two annotators preferred one passage, while the third annotator preferred the other passage. Here, the preferences for PIM and baseline were exactly 50%. This suggests that sometimes identifying a passage that summarizes the original contribution is quite difficult. When this is not the case, however, the PIM outperforms the baseline quite substantially with 70% preference.

## 4.5 Discussion and Future Work

While the Passage Impact Model provides a generative model of diachronic corpora and the relationships between individual documents, the model is still quite simple. For example, it is based on unigram models of text production. In modeling the probability of  $W^{(i)}$ , one could instead use a more sophisticated sequence model, or at least  $n$ -gram language models. Such information may help to identify coherent original ideas. Another limitation is that the model is constrained to evaluate only a small number of candidate  $Z^{(i)}$  for efficiency reasons. Developing pruning criteria is a promising direction for substantially increasing the scope of  $Z^{(i)}$  in hopes of finding better descriptions of original contributions.

Other information available for some corpora could be integrated into the model as well. For example, if citation information is available, it could provide additional constraints on the parameters during inference. Citations could be used as priors for mixing weights, modeling that documents copy primarily from those documents they cite. This could improve the accuracy of the model, and it could improve efficiency of the optimization since many mixing weights could be fixed at zero.

A more general direction for further work lies in the integration of originality de-

tection with models for idea flow. The goal is to have a unified probabilistic model that identifies the dependency structure of the corpus, with ideas originating in some documents and then flowing through the corpus. Treating these inference problems separately seems suboptimal.

## **4.6 Summary**

We have proposed an unsupervised generative model for diachronic text corpora that provides a formal structure for the process by which authors form new ideas and build on existing ideas. The model captures both novelty and impact, defining an (important) original contribution as a combination of both. For this Passage Impact Model, we have proposed an inference procedure to identify the most original passage of a document. Under reasonable approximations, the inference procedure reduces to multiple convex programs that can be solved efficiently. The method is evaluated on synthetic and real data, and it is shown to significantly outperform a heuristic baseline for selecting a passage describing the original contribution in the domains of online discussions and research articles.

## CHAPTER 5

### CONCLUSIONS AND DIRECTIONS FOR FURTHER RESEARCH

This thesis has addressed methods for helping people understand the development of ideas in collections of time-stamped text documents. The modeling and evaluation led to the following conclusions and ideas for future work.

#### 5.1 Conclusions

We have proposed and evaluated methods for helping people understand the interactions between documents and their ideas in self-referential document archives. These offline methods were based on unsupervised generative models for such document archives where documents accumulate over time and new documents refer to ideas from existing documents. For wide applicability, these methods analyzed exclusively the text of the documents to uncover the idea structure hidden in the document text. One major assumption of this work is that as authors write documents, their ideas are encoded recoverably in statistical properties of document text. We have called this set of tasks Information Genealogy. By this term, we mean in a sense that these methods can trace the textual “signatures” of ideas as they are passed from document to document over time.

In developing methods to identify the idea structure, this work has addressed tasks related to three specific concepts – influence, novelty, and original contributions. The method for detecting influence uses the Likelihood Ratio Test to analyze pairs of documents to determine whether one influenced the other. This information can be aggregated to find the most influential documents in the corpus. Then, the novelty method applied KL-Divergence from information theory to identify the documents that have the



most different ideas when compared against previously-existing documents at that time. We found, however, that novelty detection was not the correct task for finding the important new ideas that shaped the corpus. Consequently, we proposed the Passage Impact Model (Model 4.1) that combines novelty and impact for identifying these important new ideas. Within this big picture of tasks related to identifying the idea structure, we have proposed specific methods to analyze text to identify the following:

- Inter-document influence relationships that mark idea flows
- A ranking of the most influential documents
- A ranking of the most novel documents
- The most novel terms from novel documents
- The portions of documents that express important original ideas

We evaluated these methods both on synthetic data and real data, primarily from the domain of research publications. For the original contributions method, we additionally evaluated on data collected from web discussion boards. The methods were compared against text similarity baselines based on TFIDF. These methods overall were effective when compared against text similarity, especially for detecting inter-document influence and selecting important original passages.

## **5.2 Future Work**

The experiments brought to light some ideas for further study in future work, including these specific directions.

### **5.2.1 More Sophisticated Language Models**

The probabilistic language models that we considered were unigram multinomial distributions of words. While unigram language models have been successfully applied to many tasks, such as text classification and information retrieval, the independence assumption for unigram multinomial distributions is obviously not true. Authors do not actually write documents by drawing words independently from unigram multinomial distributions.

To relax this assumption, a first step would be to use more expressive language models such as  $n$ -gram language models or another model where probabilities depend on word order. More sophisticated language models could potentially take advantage of this information to do better at detecting influential documents, novel documents, original content, and so forth. In pursuing this direction, sparseness will increase with more complex language models, so that smoothing would become much more important.

### **5.2.2 Integration of Non-Textual Data**

Another obvious improvement is to integrate non-textual data into the methods. For example, the citation structure for research publications or the hyperlink structure for other document collections contains information about the influence structure between documents. While sometimes citations do not denote influence since people do cite for other reasons (), there are many cases where citations and other hyperlinks (e.g., on the Internet) do signify that there is influence of one document on another. In these cases where influence is made explicit, it makes sense to use this information in addition to text.

There are several ways to combine text and hyperlink data. In the influence method, citations could be used to select the sets of candidate documents. Or, a single generative model could describe both citations and text, so that the likelihood could simultaneously represent influence according to both text and citations. For the original passages, it might be possible to leverage the citation structure to further constrain parameters in inference. One current limitation of the Passage Impact Model is that considering all passages of text in the document is inefficient. Citations could be used to focus on the text that is more likely to be an original contribution. Within a document, perhaps text appearing in a citation context is less likely to be an original contribution, while text that is similar to the text in citation contexts from future documents that cite this document might be more likely to be an original contribution. A document's citations could also be used to select the documents that are used to explain the content of that document. Using the citations as prior information in this optimization could focus the method on the important background documents, since one of the limitations of the method is currently the number of background documents that can be used in the optimization. Overall, it seems there are many possibilities for using citations or other hyperlink data besides the text.

### **5.2.3 Unified Model for Idea Structure Inference**

Besides these obvious modeling extensions, the high-level goal is to build a unified generative model for identifying the idea structure and idea flows that describe how document collections develop. Such a model could simultaneously consider not only novelty in identifying new ideas and influence in idea flows between documents, but also other important points in the idea structure of a corpus, such as marking where an idea ends with the last document to consider it, or identifying the points where a document com-

bines two or more existing ideas into a single idea that then becomes influential, or in finding survey papers that tie together many sources of information in a conclusion. The influence, novelty, and originality models in this thesis present a start to helping people understand the idea structure of a corpus, but there is much work to be done in developing more general models that can simultaneously capture all of this development. The advantage is that instead of performing inference on each of these tasks independently, by conducting simultaneous inference in a more general model, perhaps the inference procedure could take advantage of similarities or interaction between the tasks, e.g., influence and originality, to learn a better overall solution. On the other hand, the challenge is that as more layers are added and as one tries to uncover more complicated idea structures, inference becomes trickier. For example, when combining influence and novelty in the original contributions method, the inference problem became much more complicated. Unifying the model is a good general direction for future work.

## **5.2.4 Evaluation and Collecting Data**

On the experimental side, evaluating these models is difficult because of the lack of labeled data for these tasks. For the influence method, we were able to use citations to evaluate the method, even though as we showed, the citations are not really a gold standard because of how people cite. For the original passages, collecting labeled data was even more complex, but the methods were evaluated on some real data from Slashdot discussion boards, as described in Ch. 4.

Even with the relative scarcity of labeled data, we still have evaluated the methods on synthetically-generated data derived from research publications. Synthetic data has obvious advantages, in that it is possible to control the generative setup and the data

it produces, so that one can evaluate various aspects of the models. However, since the synthetic data is generated according to the models, while the assumptions seem plausible, it is not really possible to test the modeling assumptions, as would be the case with real data. To conduct a more in-depth evaluation, more good-quality real data would be very helpful. In the future, to strengthen the evaluation, we could collect more real data or perhaps conduct user studies if there is no such data available.

### **5.2.5 Generalization to Other Domains**

In conducting the evaluation, we also want to explore these methods on other domains besides research publications. The domain of research publications is particularly convenient for several reasons, namely, that it follows the archival process where authors refer to previous influential documents, that there are collections of these documents such as NIPS freely accessible online, that it has in fact undergone much temporal development in the documents that we analyzed, and that there is explicit citation information available by which one can evaluate influence. For these reasons, much of the evaluation in this thesis has concentrated on research publications, and especially on NIPS and synthetic data generated according to NIPS documents.

We did explore data from the web discussion board Slashdot for the original contribution methods. In that setting, the method was able to successfully identify the sentences that users quoted in the discussion on web articles. The future direction is to explore how well these methods work in other domains, such as news articles, blogs, email, and so forth, or even some combination of these domains, such as how news articles influence blogs.

## 5.2.6 Scalability and Efficiency

Exploring these methods in other domains brings up issues of scalability and efficiency. In solving the optimization problems, these methods generally compute a score for some text, whether it be a document, a passage, or a set of future documents, according to the likelihood that text was generated by some mixture of existing influential documents. One limitation is in restricting the set of documents to use for computing the optimizations. For research publications, we typically used 100 previous influential documents. The methods do not scale well to very large document collections because the optimization problems grow quite large. On the other hand, with too few documents, the methods cannot correctly identify which ideas have come from existing documents. One possible solution is to use other data if available, e.g., a hyperlink structure to select the set of previous documents.

## BIBLIOGRAPHY

- Agosti, M., & Allan, J. (1997). Introduction to the special issue on methods and tools for the automatic construction of hypertext. *Information Processing and Management*, 33, 129–131.
- Agosti, M., & Crestani, F. (1993). A methodology for the automatic construction of a hypertext for information retrieval. *Proceedings of the ACM/SIGAPP Symposium on Applied Computing* (pp. 745–753).
- Agosti, M., Crestani, F., & Melucci, M. (1997). On the use of information retrieval techniques for the automatic construction of hypertext. *Information Processing and Management*, 33, 133–144.
- Allan, J. (1995). *Automatic hypertext construction*. Doctoral dissertation, Cornell University.
- Allan, J. (1997). Building hypertext using information retrieval. *Information Processing and Management*, 33, 145–159.
- Allan, J., Carbonell, J., Doddington, G., Yamron, J., & Yang, Y. (1998a). Topic detection and tracking pilot study: Final report. *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*.
- Allan, J., Papka, R., & Lavrenko, V. (1998b). On-line new event detection and tracking. *Proceedings of the SIGIR Conference on Research and Development in Information Retrieval* (pp. 37–45).
- Aya, S., Lagoze, C., & Joachims, T. (2005). Citation classification and its applications. *Proceedings of the International Conference on Knowledge Management (ICKM)*.
- Baird, L. M., & Oppenheim, C. (1994). Do citations matter? *Journal of Information Science*, 20, 2–15.

- Blei, D., Griffiths, T., Jordan, M., & Tenenbaum, J. (2003a). Hierarchical topic models and the nested chinese restaurant process. *Proceedings of the Conference on Advances in Neural Information Processing Systems (NIPS)*.
- Blei, D., & Lafferty, J. (2005). Correlated topic models. *Proceedings of the Conference on Advances in Neural Information Processing Systems (NIPS)*.
- Blei, D., Ng, A., & Jordan, M. (2003b). Latent dirichlet allocation. *Journal of Machine Learning Research (JMLR)*, 3, 993–1022.
- Blei, D. M., & Lafferty, J. D. (2006). Dynamic topic models. *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 113–120).
- Casella, G., & Berger, R. L. (2002). *Statistical inference*, chapter 10.3.1 Asymptotic Distribution of LRTs, 488–492. Duxbury.
- Coombs, J. H. (1990). Hypertext, full text, and automatic linking. *Proceedings of the SIGIR Conference on Research and Development in Information Retrieval* (pp. 83–98).
- Dietz, L., Bickel, S., & Scheffer, T. (2007). Unsupervised prediction of citation influences. *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 233–240).
- Furuta, R., Plaisant, C., & Shneiderman, B. (1989). A spectrum of automatic hypertext constructions. *Hypermedia*, 1, 179–195.
- Garfield, E. (1955). Citation indexes to science: A new dimension in documentation through association of ideas. *Science*, 122, 108–111.
- Garfield, E. (1972). Citation analysis as a tool in journal evaluation. *Science*, 178, 471–479.
- Garfield, E. (2003). The meaning of the impact factor. *International Journal of Clinical and Health Psychology*, 3, 363–369.



- Ginsparg, P. (1991). The physics e-print arxiv. <http://www.arxiv.org>.
- Griffiths, T., & Steyvers, M. (2002). A probabilistic approach to semantic representation. *Proceedings of the Annual Conference of the Cognitive Science Society*.
- Griffiths, T., Steyvers, M., Blei, D., & Tenenbaum, J. (2004). Integrating topics and syntax. *Proceedings of the Conference on Advances in Neural Information Processing Systems (NIPS)*.
- Guha, R., Kumar, R., Sivakumar, D., & Sundaram, R. (2005). Unweaving a web of documents. *Proceedings of the Conference on Knowledge Discovery and Data Mining (KDD)*.
- Havre, S., Hetzler, B., & Nowell, L. (2002). Themeriver: In search of trends, patterns, and relationships. *IEEE Transactions on Visualization and Computer Graphics*.
- Hofmann, T. (1999). Probabilistic latent semantic analysis. *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*.
- Jelinek, F. (1998). *Statistical methods for speech recognition*, chapter Basic Language Modeling, 57–78. MIT Press.
- Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46, 604–632.
- Kleinberg, J. (2002). Bursty and hierarchical structure in streams. *Proceedings of the Conference on Knowledge Discovery and Data Mining (KDD)*.
- Kolenda, T., Hansen, L. K., & Larsen, J. (2001). Signal detection using ICA: Application to chat room topic spotting. *Proceedings of the International Conference on Independent Component Analysis and Signal Separation (ICA)* (pp. 540–545).
- Krause, A., Leskovec, J., & Guestrin, C. (2006). Data association for topic intensity tracking. *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 497–504).

- Kurland, O., & Lee, L. (2004). Corpus structure, language models, and ad hoc information retrieval. *Proceedings of the SIGIR Conference on Research and Development in Information Retrieval* (pp. 194–201).
- Kurland, O., & Lee, L. (2006). Respect my authority! hits without hyperlinks, utilizing cluster-based language models. *Proceedings of the SIGIR Conference on Research and Development in Information Retrieval* (pp. 83–90).
- Lavrenko, V., Allan, J., Deguzman, E., Laflamme, D., Pollard, V., & Thomas, S. (2002). Relevance models for topic detecting and tracking. *Human Language Technology*, 104–110.
- Lelu, A. (1991). Automatic generation of hypertext links in information retrieval systems: A stochastic and an incremental algorithm. *Proceedings of the SIGIR Conference on Research and Development in Information Retrieval* (pp. 326–336).
- Li, W., & McCallum, A. (2006). Pachinko allocation: Dag-structured mixture models of topic correlations. *Proceedings of the International Conference on Machine Learning (ICML)*.
- MacRoberts, M. H., & MacRoberts, B. R. (1989). Problems of citation analysis: A critical review. *Journal of the American Society for Information Science*, 40, 342–349.
- Mann, G., Mimno, D., & McCallum, A. (2006). Bibliometric impact measures leveraging topic analysis. *Proceedings of the Joint Conference on Digital Libraries (JCDL)*.
- Manning, C. D., & Schuetze, H. (1999). *Foundations of statistical natural language processing*. MIT Press.
- McCallum, A., Corrada-Emanuel, A., & Wang, X. (2005). Topic and role discovery in social networks. *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*.

- Mei, Q., Ling, X., Wondra, M., Su, H., & Zhai, C. (2007). Topic sentiment mixture: Modeling facets and opinions in weblogs. *Proceedings of the World Wide Web Conference (WWW)* (pp. 171–180).
- Mei, Q., & Zhai, C. (2005). Discovering evolutionary theme patterns from text - an exploration of temporal text mining. *Proceedings of Conference on Knowledge Discovery and Data Mining (KDD)*.
- Mei, Q., & Zhai, C. (2008). Generating impact-based summaries for scientific literature. *Proceedings of the Association for Computational Linguistics (ACL)* (pp. 816–824).
- MOSEK (2008). <http://www.mosek.com/index.html>.
- NIPS Online (2000). The Text Repository. <http://nips.djvuzone.org/txt.html>.
- NIST (2001). Document understanding conferences. <http://duc.nist.gov/>.
- Osareh, F. (1996). Bibliometrics, citation analysis and co-citation analysis: A review of literature i. *Libri*, 46, 149–158.
- Page, L., Brin, S., Motwani, R., & Winograd, T. (1998). *The pagerank citation ranking: Bringing order to the web* (Technical Report). Computer Science Department, Stanford University.
- Salton, G., & Buckley, C. (1988). Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24, 513–523.
- Salton, G., & Buckley, C. (1991). Automatic text structuring and retrieval – experiments in automatic encyclopedia searching. *Proceedings of the SIGIR Conference on Research and Development in Information Retrieval* (p. 21).
- Shaparenko, B., Caruana, R., Gehrke, J., & Joachims, T. (2005). Identifying temporal patterns and key players in document collections. *Proceedings of the IEEE ICDM Workshop on Temporal Data Mining: Algorithms, Theory and Applications (TDM)* (pp. 165–174).

- Soboroff, I., & Harman, D. (2003). Overview of the TREC 2003 novelty track. *Proceedings of the Text Retrieval Conference (TREC)*.
- Steyvers, M., Smyth, P., Rosen-Zvi, M., & Griffiths, T. (2004). Probabilistic author-topic models for information discovery. *Proceedings of the Conference on Knowledge Discovery and Data Mining (KDD)* (pp. 306–315).
- Swan, R., & Jensen, D. (2000). Timemines: Constructing timelines with statistical models of word usage. *Proceedings of the KDD Workshop on Text Mining* (pp. 73–80).
- Wang, X., Li, W., & McCallum, A. (2006). A continuous-time model of topic co-occurrence trends. *AAAI Workshop on Event Detection*.
- Wang, X., & McCallum, A. (2006). Topics over time: A non-markov continuous-time model of topical trends. *Proceedings of the Conference on Knowledge Discovery and Data Mining (KDD)* (pp. 424–433).
- Wilkinson, R., & Smeaton, A. F. (1999). Automatic link generation. *ACM Computing Surveys (CSUR)*, 31, 27.
- Zhai, C. (2002). *Risk minimization and language modeling in information retrieval*. Doctoral dissertation, Carnegie Mellon University.
- Zhai, C., & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22, 179–214.