Finding Expert Authors in Institutional Repositories

Miles Efron
Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign
501 East Daniel St.
Champaign, IL 61820
mefron@gmail.com *

June 4, 2009

Abstract

Institutional repositories collect the output of scholarly activities at universities and similar organizations. While a great deal of research has treated curation, collection development, and interoperability in institutional repositories, the literature regarding information retrieval in these repositories is small. This poster reports initial work on an ongoing project to improve access to institutional repository data through advanced information retrieval techniques. In particular, this poster treats the problem of "expert finding" in institutional repositories. We propose that a useful mode of information discovery lies in supporting queries seeking people (as opposed to documents) who evince expertise on a given topic. This poster describes a framework for conducting searches of expert authors in institutional repositories. Additionally we report results from several empirical tests designed to judge the effect of various modeling decisions in the expert finding problem for expert finding in institutional repositories.

1 Introduction

Institutional repositories "organize, preserve, access and facilitate use of digital content produced by members of their communities" [12, p. 1]. Though their missions vary, the majority of institutional repository projects serve university communities (cf. [4]). This poster describes an ongoing project to improve access to the information stored in institutional repositories. While the larger project aims to improve access by applying advanced information retrieval (IR) techniques to repository data, this poster focuses on a particular problem: finding expert authors in a particular domain within a repository.

While institutional repositories have an obvious role in collecting and preserving data, they bear an additional responsibility insofar as repositories ensure access to these data. Clifford Lynch has argued that "a university-based institutional repository is a set of services that a university offers to the members of its community for the management and dissemination of digital materials...." [9, p. 328]. The argument that informs this research is that a service that is of prime importance in institutional repositories is information retrieval. We argue that supporting *useful* methods of search, retrieval, and discovery of information is essential to the success of these repositories. By improving retrieval in repositories, we argue, we stand to increase the value of repository projects.

Of course most repositories provide a basic search facility (keyword, author, title, etc.). But this study is part of an ongoing project to introduce advanced, imaginatively applied IR to institutional repositories. In this paper we focus on a single problem: helping users find authors who are experts in a particular topic. The idea is to support queries such as *which scholars at this institution are experts in the topic x*? Additionally,

*Portions of this work were completed when the author was an assistant professor in the School of Information at the University of Texas at Austin.

in a federated repository environment (i.e. by combining data from multiple repositories), a system based on the models presented here could help people find universities that have expertise in a particular field.

2 Problem Statement

Our topic falls under the umbrella of a sub-problem within information retrieval: so-called expert finding [3, 14, 15, 2]. Expert finding has emerged in contexts such as enterprise search and retrieval over email forums. In work similar to ours, a recent paper pursues the expert finding problem in the context of a university website [7].

In these contexts, it is useful to consider a *person* as the unit of retrieval, instead of a document. Institutional repositories traffic in documents (preprints, data sets, learning objects, etc.). This paper collects each author's document representations to build a representation of that person (or agent's) authorial expertise. The goal of the work is to help people discover which members of the university community are expert on a topic x. This question is of interest to the general public, for instance, when journalists seek experts on a subject for a story. Additionally, we argue that a federated search over many repositories would be of interest to scholars who are interested in finding authors with expertise in a particular topic area.

Modeling author expertise for expert finding in institutional repositories raises a number of challenges. First, we take up the problem of data representation. How should we represent documents, queries and people in order to allow meaningful searches for experts? This question leads naturally into the question: what retrieval method provides effective, well-motivated search over a database of expert authors? Finally, we ask how the particular structure of repository data differentiates expert finding in this domain, as opposed to traditional, ad hoc retrieval. Additional questions are laid out in Section 5.

3 Motivation and Approach

Expert finding is a relatively new problem in information retrieval, and the best methods for finding experts is an open question (cf. the TREC enterprise track[1]). In the context of institutional repositories, two broad strategies lend themselves to the expert finding problem. Elsewhere in the literature (cf. [1, 2]) authors articulate variations on this strategic division as the distinction between "author-" and "document-centric" approaches to expert finding.

3.1 Author-centered Language Modeling for Expert Search

Our first way to perform expert search involves creating an explicit model of each author in the document collection. Each author in the collection is thus considered a candidate expert, and retrieval simply involves ranking candidates in decreasing likelihood (in the informal sense) of their being experts on a given topic. Probabilistic language models provide a natural, well-motivated form for these author representations.

Probabilistic approaches based on statistical language modeling form a mainstay of contemporary information retrieval [11, 19, 20]. Language modeling has also proven useful in the context of expert finding [1]. Though Section 4 describes other methods of retrieval, our main focus will be on applying language modeling techniques to finding expert authors in institutional repositories.

In information retrieval a language model is a statistical distribution over an indexing vocabulary of m unique terms. Typically we use the multinomial distribution, which is characterized by a parameter vector $\vec{\theta}$, where θ_j is the probability that the model generates term j.

In our approach we imagine that a language model μ_i corresponds to each author a_i in the repository of document D. For a particular author a_i the parameter $\theta a_i j$ is the probability that this author writes the jth term in a document.

[1] `http://www.ins.cwi.nl/projects/trec-ent/wiki/, downloaded June 1, 2009`

Given a query q represented by a vector of term counts $\vec{q}$, we may rank authors in decreasing order of the likelihood that their respective models generated $\vec{q}$. This is the basic "query generation" approach to retrieval.

In this work we use a slight variation on language modeling–the Kullback-Leibler divergence model, as described in [18, 20][2]. Here we assume that the query q is also generated by a language model. Documents are then ranked in increasing order of the Kullback-Leibler divergence between their own models and the query model (see [5] for background on the Kullback-Leibler divergence). The KL divergence between two discrete distributions X and Y is

$$D(X||Y) = \sum p(x) \log \frac{x}{y} \tag{1}$$

where the sum is taken over the outcome space (here over all terms in the vocabulary).

3.1.1 Model Estimation

The maximum likelihood estimator (MLE) for term t in the language model of document d is simply the frequency of t in d divided by the length of d. The MLE for the query model and each document model may then be plugged into Eq. 1 for retrieval.

However, in practice, the MLE is typically a poor estimator of word probability. A key operation in language model IRet is *smoothing*, an effort to improve upon the MLE in the models used in document ranking. A full discussion of language model smoothing is given in [19]. We discuss two methods of smoothing in this paper: Bayesian updating with Dirichlet priors and Jelinek-Mercer smoothing. In the experiments that follow, the Dirichlet smoothing parameter $\mu = 1000$ and Jelinek-Mercer's parameter $\lambda = 0.1$.

Aside from smoothing, however, another problem faces us in the context of expert finding: how should we estimate the language models for authors? In an institutional repository, each author's text is distributed over possibly many documents. From these (and any additional data) we wish to induce a single language model per author.

To accomplish this we propose a simple approach: each author's model is calculated by concatenating all of the text from his or her records in the repository. Thus the maximum likelihood estimator for the probability that author j uses word i is

$$Pr(w_i|a_j) = \frac{n(w_i,\ a_j)}{n(a_j)} \tag{2}$$

where $n(w_i,\ a_j)$ is the number of times *across the collection* that author j uses word i and $n(a_j)$ is the total number of words (tokens) used by the author.

In other words each author is represented by a "pseudo document," a string of text obtained by concatenating all of his or her contributions to the repository. During retrieval, we rank authors in increasing order of the estimated KL divergence between their language model and the query model.

3.2 Document-Centered Expert Finding

In our previous discussion we relied on an explicit model of each author in the collection. Earlier work has suggested an alternative, document-centric way to rank authors, and here we pursue such a strategy. The idea is to perform a standard document retrieval on our repository documents as a first step. We then undertake a second step during which we iterate over the top n results from our document search[3]. During this iteration, we record for each author i the estimated relevance of documents that he or she created. The author's final relevance score is simply the sum of his or her documents' relevance scores.

Stating things a bit more formally, let $f(q, d)$ be an arbitrary retrieval function that takes a query q and a document d and retrieves a real-valued "relevance" estimate for the pair. We shall call $f(q, d)$ the retrieval

[2]As implemented here the KL retrieval method gives the same ranking as the query likelihood model. We use KL because it is more amenable to incorporating relevance feedback, a project we plan to undertake in future work.

[3]In this study we set $n = 50$ after lengthy empirical testing. This value yielded the best results over a wide range of n.

status value (RSV) of d with respect to q. $f(.)$ might be, for instance, the KL divergence formula given above, the cosine similarity measure, or the RSV from the Okapi BM25 algorithm [13].

The document-centric approach described in this section is similar in motivation to the voting model proposed in [10]. In future work we plan to situate this document-centric approach in a form that is analogous to the language modeling strategy outlined in the previous section.

4 Empirical Analysis

Because this paper describes preliminary work, we focus on results that give impressions on three broad research questions:

1. Does the language modeling approach outlined in Section 3 yield better retrieval performance than another state-of-the-art retrieval model?
2. Which approach–author- or document-centric–gives more effective expert search over repository data?
3. Does institutional repository data present different challenges than data in either other expert finding problems or in standard ad hoc IR?

In Section **??** we discuss several additional questions raised by the present work.

4.1 Test Data

To address our research questions we constructed two test collections. Each collection consisted of data from a different repository. But perhaps more importantly, the queries and relevance judgments were different in each collection.

Table 1: Data sets used in the experiments reported in this paper.

Corpus	# Docs	# People	# Queries	Query Type	# Experts
UIUC	10,425	7,457	20	General	221
MIT	24,810	31,292	20	Technical	44

Summary statistics for both test collections appear in Table 1. The following paragraphs describe the data in detail.

Both collections consisted of Dublin Core metadata harvested using the Perl open source Open Archives data harvester `Net::OAI::Harvester`[4]. Thus only unqualified Dublin Core (as per the Open Archives Initiative protocol for metadata harvesting[5], downloaded June 1, 2009.) was fetched for each record [6].

Documents were indexed using the `Lemur` toolkit[6]. We used no stemming nor a stoplist. Additionally, we omitted any sort of disambiguation of personal names. Name disambiguation has been shown to improve retrieval significantly [17]. However, we omitted this step since our goal here is simply comparison of underlying models, not high performance per se. Thus any character string listed in a Dublin Core *creator* or *contributor* element was considered as an "author" of a document for the purposes of this study.

For each record we indexed any *title*, *subject*, *description*, and *coverage* elements. In future work we plan to investigate whether the choice of elements indexed bears on retrieval performance.

The first corpus consists of all Dublin Core records fetched using the `ListRecords` OAI verb on the IDEALS repository at the University of Illinois[7] (UIUC). The second collection consists of records retrieved

[4] `http://search.cpan.org/ esummers/OAI-Harvester-1.0/lib/Net/OAI/Harvester.pm`, downloaded June 1, 2009.
[5] `http://www.openarchives.org/pmh/`
[6] `http://lemurproject.org`, downloaded June 2, 2009
[7] `http://www.ideals.uiuc.edu/`, downloaded June 1, 2009.

from the institutional repository at MIT[8]. Because this is a pilot study we opted not to burden the MIT OAI server and thus requested only those documents submitted during 2008 through May 2009. These repositories were selected because they are both well established and thus contain a realistically large collection of documents of various types (e.g. theses, pre-prints, learning objects).

Creating queries and relevance judgments for each collection involved several admittedly subjective decisions. These decisions hinged on the larger question: in the context of expert finding, what does it mean for someone to be an expert on a topic?

In forming queries this question raised the matter of information needs. If people used a repository expert finder, what kinds of questions would they be trying to answer? Relevance judgments present a similar problem. Presumably searchers with different levels of expertise, for instance, might bring different interests to the expert finding problem.

To handle these issues we created two sets of queries, applied each to one of the repository corpora and then performed manual relevance judgments as described below. Our two query types are:

- *General*: Queries that a person such as a journalist might have during the writing of an article for a non-expert audience
- *Technical*: Queries indicative of a user who is him- or herself expert on the topic at hand.

Operationalizing these query types led to a methodology for creating queries and for deriving relevance judgments.

The technical queries are simple to describe. Here we started with the assumption that a searcher is interested in a very narrow area of expertise. Thus *information retrieval* would not be an appropriate technical query, but *language model smoothing in information retrieval* would be. To build these queries and relevance judgments we used doctoral dissertations from the MIT repository. We collected the 20 most recently submitted dissertations and (manually) constructed a query based on the content of the dissertation title and abstract.

Relevant experts for each technical query were simply taken to be the dissertation author and the members of his or her committee. Thus there were approximately four or five experts for each technical query. Obviously this method of defining ground truth ignores the possibility that other experts at MIT could exist (and that occasionally non-experts serve on a committee).

Our non-technical queries followed a different logic. Our goal with these queries was to field a more general information need. Hence our model was a journalist interested in a topic x as he writes an article about it. This person, we assume, is articulate and resourceful, but not an expert himself. The putative goal of these queries is to locate someone who can speak with authority on the topic.

To build these queries we took two strategies. First, for fifteen queries we started by issuing a web search on Google for the phrase *professor of * at * University* in the archives of *the New York Times*. From these results we found fifteen *Times* articles in which a professor or graduate student researcher was interviewed for his or her expertise on the subject of the article. We then manually crafted a keyword query similar to the topic discussed by the interviewee. Five additional queries were created manually by examining which subject headings in the UIUC repository were most widely used. We did this in efforts to find topics with many experts.

Relevance judging for the general queries was conducted by thorough examination of the collection. For each test query we manually conducted many related queries using the repository's native search interface (not our own). We also traversed the site's subject headings in efforts to find documents related to each topic. As we searched, we noted each author or contributor associated with documents deemed relevant to the query. These authors were judged to be experts on the topic if a search for their name (as author) and the query itself on Google Scholar yielded results with at least 10 (an admittedly arbitrary number) citations.

[8] `http://dspace.mit.edu/`, downloaded June 1, 2009.

4.2 Experimental Results

Tables 2 and compare retrieval performance using each method of representation–person- and document-centric. The topmost headings describe three different retrieval strategies. For these experiments we used language modeling with both Dirichlet and Jelinek-Mercer (JM) smoothing, as well as the BM25 Okapi approach. The remaining column headings report three performance metrics averaged over all 20 queries for each corpus: mean average precision (MAP), precision at five documents retrieved (P@5), and recall. Retrievals returned at most the top 1000 documents for each query.

Our first research question pursues that matter of retrieval effectiveness using different retrieval models. The language modeling approach outlined in Section **??** has an intuitive as well as a theoretical appeal. But as we can see from Tables 2 and there appears to be no compelling "winner" between the language modeling approach and our baseline BM25 runs.

Our second research question deals with the problem of expert representation. Should we model our data in a document- or person-centric way?

Table 2: Baseline performance for person-centric expert finding over UIUC and MIT data sets.

	Dirichlet			JM			Okapi		
Corpus	MAP	P@5	Rec.	MAP	P@5	Rec.	MAP	P@5	Rec.
UIUC	0.477	0.495	0.846	0.339	0.400	0.846	0.392	0.457	0.846
MIT	0.611	0.260	0.818	0.678	0.310	0.841	0.643	0.310	0.841

Table 3: Baseline performance for document-centric expert finding over UIUC and MIT data sets.

	Dirichlet			JM			Okapi		
Corpus	MAP	P@5	Rec.	MAP	P@5	Rec.	MAP	P@5	Rec.
UIUC	0.381	0.429	0.747	0.315	0.419	0.747	0.359	0.419	0.706
MIT	0.262	0.180	0.773	0.364	0.270	0.773	0.121	0.040	0.773

Comparing Tables 2 and 3 it is quite clear that the person-centric approach to expert finding outperforms the document-centric approach pursued here. All differences between runs using person- and document-centric approaches in Tables 2 and 3 are statistically significant ($p < 0.01$) using a one-sided Wilcoxon rank sum test. This is somewhat surprising as the method of summing scores in a document-centric system has performed well in other studies.

Another important result apparent in Tables 2 and 3 is the difference in performance obtained using our three retrieval models, Dirichlet- and JM-smoothed language models and Okapi weighting. Focusing on the UIUC data in Table 2 we see that Dirichlet smoothing gives decisively superior performance (on MAP and P@5) than the other two methods. However, in the same table we see that this is not the case for the MIT data, where JM and Okapi outperform the Dirichlet-smoothed language modeling approach. A similar dynamic is at work in Table 3.

This result speaks to our third research question: does expert finding over institutional repository data present unique challenges? The discrepancy between The Dirichlet and Jellinek-Mercer-smoothed language models is unusual. It is the case that Dirichlet smoothing often gives better results than other smoothing methods [19, 8]. But such wide variation in performance is atypical. Moreover, the fact that Dirichlet smoothing performs much *better* than JM for the UIUC data, but much *worse* for the MIT data is perplexing. This result suggests that the data (either our corpora, queries, or relevance judgments) are quite different in this experiment than they are in standard retrieval.

Why does Dirichlet do so much better than JM and Okapi on UIUC but not on MIT? A possible explanation lies in the matter of document length. Smucker and Allen have argued that a principal asset of

Dirichlet smoothing is that it gives and implicit advantage to longer documents [16], which at least in TREC collections, have a relatively high likelihood of relevance.

An interesting possibility for the results in Tables 2 and 3 with respect to Dirichlet's superiority is that Bayesian smoothing's favor for longer documents rewards authors who exhibit either or both of two behaviors in their repository submissions:

- creating long, descriptive metadata
- submitting many items to the repository.

Since our author models consist of text concatenated from each author's various contributions, either of these behaviors would lead a particular author to have a comparatively long "pseudo-document."

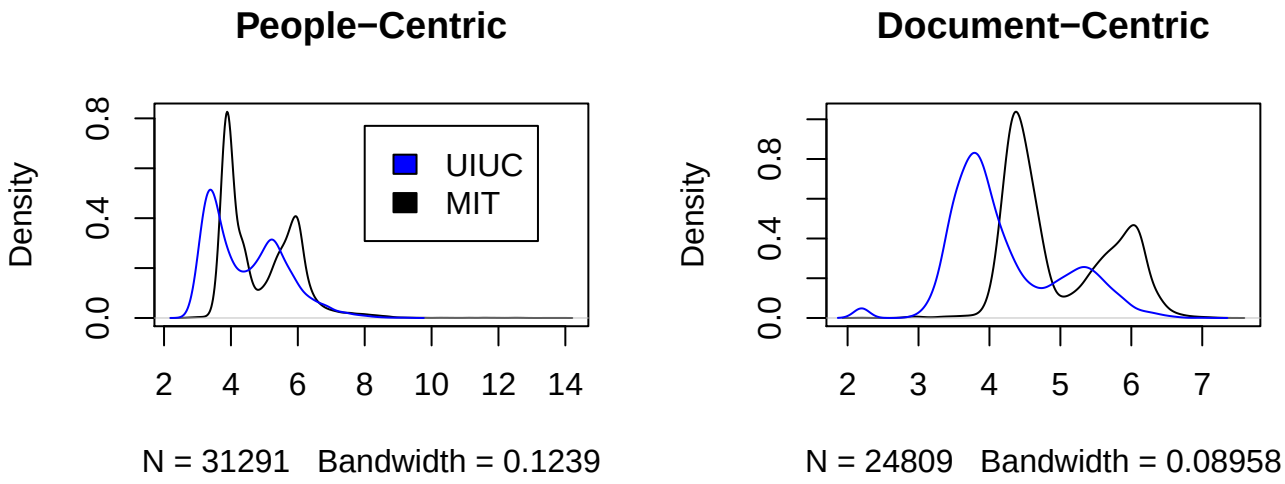

Figure 1: Density plots for person-centric pseudo-documents and for regular documents over UIUC and MIT data. Plotted densities are for the log-length of each respective document.

However, on first glance, Figure 1 seems to contradict this intuition. The figure gives density plots for each test repository; one panel plots people-centric pseudo-documents and the other plots results from the observed OAI documents. The variable in the panels is the log-length of each document. From Figure 1 it appears that on the whole, MIT's documents are longer the UIUC's, suggesting that Dirichlet's benefit in UIUC is due to something other than document length.

However, the analysis changes if we look at Figure 2. Here we have separated out the *log*-lengths for those people judged to be experts in our test collection (represented by colored X's). Although on average MIT documents are longer, the ratio of judged experts' pseudo-document length to non-expert lengths is greater for UIUC. That is, expert document lengths tend to be *relatively* longer than non-expert document lengths in UIUC, which isn't the case in MIT.

To formalize this intuition we conducted a one-sided test between binomial proportions (expert vs. non-expert) in each corpus. The test yielded $p < 0.001$, suggesting that the pseudo-document length for judged experts versus length for non-judged authors is higher in the UIUC data than in the MIT. This result lends credence to the possibility that Dirichlet smoothing is able to outperform our other models on the person-centric UIUC data by capitalizing on the difference in verbosity in expert and non-expert records.

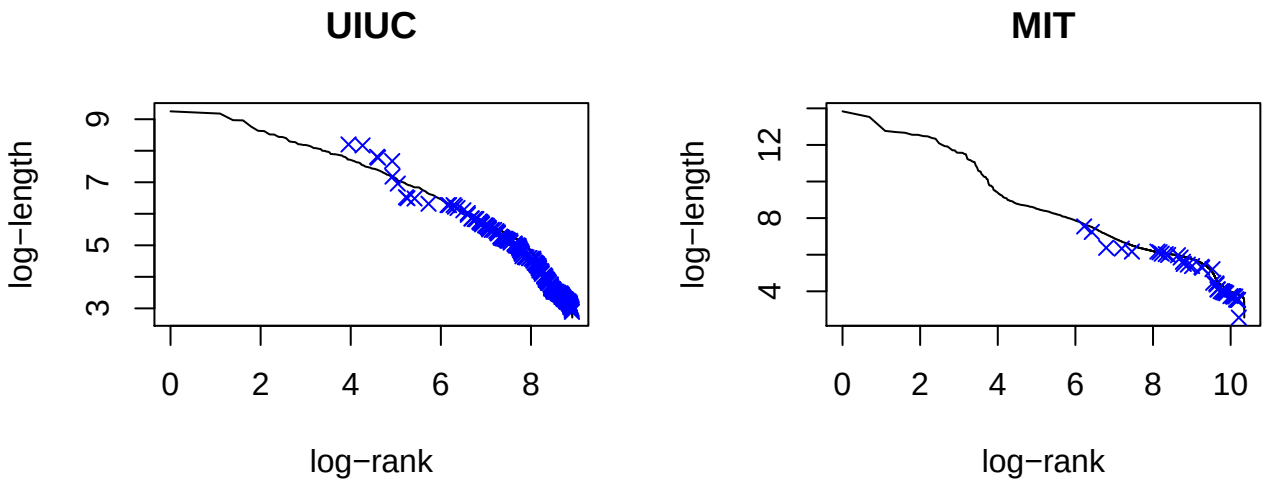

Figure 2: Person-centric pseudo-document length (log-log scale). The line shows log-length for authors not judged to be experts in the test collections. Colored X's show log-length for judged experts.

5 Conclusion and Future Directions

For institutional repositories to offer a meaningful resource to the academic community they must deliver services that are more ambitious and innovative than those offered by standard search services. In other words, for repositories to become central to scholarly communication they must improve access to scholarly output in a meaningful way. The work proposed in this poster provides a tangible step towards providing this type of innovation.

We propose that the expert finding task is useful in its own right. Moreover, we hope that in future work we may integrate services such as expert finding into access systems that increase the value of data stored in institutional repositories. Information retrieval, imaginatively applied, we argue, will add to the utility–and by extension, we hope the adoption–of institutional repositories.

This work is preliminary, but the results reported in Section 4 suggest that expert finding based on OAI metadata is a tractable problem, and that the methods presented in Section 3 provide compelling responses to this challenge.

However, we have necessarily omitted discussion of several important issues. These include:

- *Full-text data*: Our analysis relied only on Dublin Core metadata. But most repositories also store full text documents. How would these data bear on the expert finding problem?
- *Meta-information*: Institutional repository data yields several types of secondary information that could be useful for IR. For instance, could the number of contributions for each author improve our models? Perhaps trawling external databases for evidence of expertise would be useful.
- *Problem representation*: We have assumed that users would enter standard keyword queries in an expert finding system. Is this in fact a good way to proceed? Secondly, we omitted discussion of presenting results. What would a returned list of putative experts look like? How could we fashion a useful surrogate for each author in a return set?

These are exciting questions, and we look forward to answering them in future work. To pursue these answers we plan to undertake a larger, more sophisticated set of experiments using more corpora and queries, and

leveraging established methods of proper name normalization.

References

[1] Krisztian Balog, Leif Azzopardi, and Maarten de Rijke. Formal models for expert finding in enterprise corpora. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 43–50, New York, NY, USA, 2006. ACM.

[2] Krisztian Balog, Leif Azzopardi, and Maarten de Rijke. A language modeling framework for expert finding. *Inf. Process. Manage.*, 45(1):1–19, 2009.

[3] Krisztian Balog and Maarten de Rijke. Finding experts and their eetails in e-mail corpora. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 1035–1036, New York, NY, USA, 2006. ACM.

[4] Paul Conway. Modeling the digital content landscape in universities. *Library Hi Tech*, 26(3):342–354, 2008.

[5] Thomas Cover and Joy Thomas. *Elements of Information Theory*. Wiley, New York, 1991.

[6] Carl Lagoze and Herbert Van de Sompel. The Open Archives Initiative: building a low-barrier interoperability framework. In *JCDL '01: Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries*, pages 54–62, New York, NY, USA, 2001. ACM.

[7] Ruud Liebregts and Toine Bogers. Design and evaluation of a university-wide expert search engine. In *ECIR '09: Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, pages 587–594, Berlin, Heidelberg, 2009. Springer-Verlag.

[8] David E. Losada and Leif Azzopardi. An analysis on document length retrieval trends in language modeling smoothing. *Information Retrieval*, 11:109–138, 2008.

[9] Clifford A. Lynch. Institutional repositories: Essential infrastructure for scholarship in the digital age. *Libraries and the Academy*, 3(2):327–336, 2003.

[10] Craig Macdonald and Iadh Ounis. Voting techniques for expert search. *Knowl. Inf. Syst.*, 16(3):259–280, 2008.

[11] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. *Research and Development in Information Retrieval*, pages 275–281, 1998.

[12] Soo Young Rieh, Karen Markey, Beth St. Jean, Elizabeth Yakel, and Jihyun Kim. Census of institutional repositories in the U.S.: A comparison across institutions at different stages of IR development. *D-Lib Magazine*, 13(11/12), 2007. http://www.dlib.org/dlib/november07/rieh/11rieh.html, retrieved Feb. 2, 2009.

[13] S. E. Robertson, S. Walker, S. Jones, M. Hancock Beaulieu, and M. Gatford. Okapi at TREC-3. In *Proceedings of TREC-3, the 3rd Text REtrieval Conference*, pages 109–127. NIST, 1995.

[14] Pavel Serdyukov, Djoerd Hiemstra, Maarten Fokkinga, and Peter M. G. Apers. Generative modeling of persons and documents for expert search. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 827–828, New York, NY, USA, 2007. ACM.

[15] Pavel Serdyukov, Henning Rode, and Djoerd Hiemstra. Modeling multi-step relevance propagation for expert finding. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management*, pages 1133–1142, New York, NY, USA, 2008. ACM.

[16] Mark D. Smucker and James Allan. An investigation of Dirichlet prior smoothing's performance advantage. Technical Report IR-548, CIIR University of Massachusetts, Amherst, 2007. http://www.mansci.uwaterloo.ca/ msmucker/publications/SmuckerAllan-Smoothing-IR548.pdf.

[17] Vetle I. Torvik, Marc Weeber, Don R. Swanson, and Neil R. Smalheiser. A probabilistic similarity metric for medline records: A model for author name disambiguation. *JASIST*, 56(2):140–158, 2005.

[18] Chengxiang Zhai and John Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 403–410, New York, NY, USA, 2001. ACM.

[19] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 2(2):179–214, 2004.

[20] Chengxiang Zhai and John Lafferty. A risk minimization framework for information retrieval. *Information Processing and Management*, 42(1):31–55, 2006.