Leif Azzopardi Gabriella Kazai
Stephen Robertson Stefan Rüger
Milad Shokouhi Dawei Song
Emine Yilmaz (Eds.)

LNCS 5766

Advances in Information Retrieval Theory

Second International Conference on the Theory of Information Retrieval, ICTIR 2009 Cambridge, UK, September 2009, Proceedings

Springer

Leif Azzopardi Gabriella Kazai
Stephen Robertson Stefan Rüger
Milad Shokouhi Dawei Song
Emine Yilmaz (Eds.)

Advances in Information Retrieval Theory

Second International Conference
on the Theory of Information Retrieval, ICTIR 2009
Cambridge, UK, September 10-12, 2009
Proceedings

Springer

Volume Editors

Leif Azzopardi
University of Glasgow
Department of Computing Science
Sir Alwyn Williams Building, Lilybank Gardens
Glasgow, G12 8QQ, Scotland, UK
E-mail: leif@dcs.gla.ac.uk

Gabriella Kazai
Stephen Robertson
Milad Shokouhi
Emine Yilmaz
Microsoft Research Ltd
7 J.J. Thomson Avenue, Cambridge, CB3 0FB, UK
E-mail: {gabkaz, ser, milads, eminey}@microsoft.com

Stefan Rüger
The Open University
Knowledge Media Institute
Milton Keynes, MK7 6AA, UK
E-mail: s.rueger@open.ac.uk

Dawei Song
The Robert Gordon University
School of Computing
St Andrew Street, Aberdeen, AB25 1HG, UK
E-mail: d.song@rgu.ac.uk

Library of Congress Control Number: 2009934008

CR Subject Classification (1998): H.3, H.2, I.2.3, I.2.6, F.2.2, H.4, H.5.2-4, I.7

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-642-04416-6 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-04416-8 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12754809 06/3180 5 4 3 2 1 0

In memory of Sándor Dominich

Preface

These proceedings contain the refereed papers and posters presented at the Second International Conference on the Theory of Information Retrieval (ICTIR 2009), held at Microsoft Research in Cambridge, UK, September 10–11, 2009.

This biennial international conference provides an opportunity for the presentation of the latest work describing theoretical advances in the field of information retrieval (IR). The first ICTIR was held in Budapest in October 2007, organized by Keith van Rijsbergen, Sándor Dominich, Sándor Darányi, and Ferenc Kiss. ICTIR was brought about by the growing interest in the consecutive workshops run at ACM SIGIR each year from 2000 until 2005 on Mathematical and Formal Methods in IR (Athens, Greece, 2000; New Orleans, USA, 2001; Tampere, Finland, 2002; Toronto, Canada, 2003; Sheffield, UK, 2004; Salvador, Brazil, 2005). This sustained initiative was in a large part down to the determination of Sándor Dominich and his passion for all things good, formal and mathematical. The foundation and the success of ICTIR is a direct result of his commitment and dedication to fostering research and development into the theoretical underpinnings of IR. His dedication is epitomized by his two books on the subject: *Mathematical Foundations in Information Retrieval* published in 2001, and *The Modern Algebra of Information Retrieval* published in 2008. While his efforts to promoting formal methods for IR have led to the foundation of ICTIR, sadly, his untimely passing in 2008 means that he is unable to witness how the theory of IR unfolds in the future. Nonetheless, his belief in the importance of theory and his spirit in advocating the development of formal methods in IR lives on through this conference series. We dedicate ICTIR 2009 to Sándor Dominich as a tribute to his contribution to the field.

ICTIR 2009 presented the latest developments in IR and boasted a high-quality program covering a diverse range of topics. The papers accepted for publication and presentation at ICTIR 2009 were selected from a total of 82 submissions, which were received from Continental Europe (39%), UK (21%), North America (18%), Asia and Australasia (10%), Middle East and Africa (12%). The submissions were assessed by at least three reviewers in a double-blind review process, and were ranked according to their scientific quality, originality, and contribution to the theory of IR. In total, 18 full papers (22%), 14 short papers (17%), and 11 posters (13%) were accepted. We categorized the accepted contributions into four main themes: novel IR models, evaluation, efficiency, and new perspectives in IR. Twenty-one papers fall into the general theme of novel IR models, ranging from various retrieval models (8), query and term selection models (4), Web IR models (3), developments in novelty and diversity (3), to the modeling of user aspects (3). There are four papers on new evaluation methodologies, e.g., modeling score distributions, evaluation over sessions, and an axiomatic framework for XML retrieval evaluation. Three papers focus on the issue of efficiency and offer solutions to improve the tractability of PageRank, data-cleansing practices for training classifiers, and approximate search for distributed IR. Finally, four papers look into new perspectives of IR and shed light on some new emerging areas of interest, such as the application and adoption of quantum theory in IR.

We would like to thank the invited speaker, Peter Bruza, for his thought-provoking keynote speech on using quantum theory to develop a new suite of information-processing models that are motivated from a cognitive science perspective. We would also like to thank all the authors who submitted their work for consideration, and all the participants and student volunteers for their contributions and help. We are grateful to the members of the Program Committee for their time and effort in providing timely and high-quality feedback and reviews.

Finally, we would like to say special thanks to the following organizations and individuals who helped to make ICTIR 2009 a success:

– Microsoft Research for hosting the event and providing the excellent conference facilities, as well sponsoring the conference dinner. We especially thank Rachael Billing (overall organization, banquet, catering), Nick Duffield (graphics design, marketing materials), Sarah Head (marketing, conference bags), Sarah Nightingale (facilities), Fabien Petitcolas (sponsorship), Mari Ann Lindqvist (finance), Adrian Cooper (security), Ian Kelly (IT) and the entire IT support team.
– The Open University for providing conference website design, registration and financial management. Many thanks go to Damian Dadswell (Web), Harriett Cornish (initial graphical designs), and Jane Whild, Rachel Barnett, Aneta Tumilowicz and The Open University's Finance devision (budget and financial management).
– The British Computer Society - Information Retrieval Specialist Group (BCS-IRSG) for providing financial support for students and for sponsoring 40 copies of book *The Modern Algebra of Information Retrieval* as the tribute to Sándor Dominich.
– The editorial staff at Springer for their agreement and assistance in publishing the conference as part of the *Lecture Notes in Computer Science* (LNCS) series.
– Yahoo Research for sponsoring the Best Student Paper Award.
– True Knowledge for their kind sponsorship.

September 2009

Leif Azzopardi
Gabriella Kazai
Stephen Robertson
Stefan Rüger
Milad Shokouhi
Dawei Song
Emine Yilmaz

Organization

Organizing Institutions

ICTIR 2009 was organized by Microsoft Research Cambridge, the Knowledge Media Institute of the Open University, the Department of Computing Science of University of Glasgow, and the School of Computing of the Robert Gordon University.

Conference Chairs

Conference Chairs	Gabriella Kazai, Microsoft Research, UK
	Stefan Rüger, The Open University, UK
Program Chairs	Leif Azzopardi, University of Glasgow, UK
	Dawei Song, The Robert Gordon University, UK
Honorary Chair	Keith van Rijsbergen, University of Glasgow, UK
Local Chairs	Stephen Robertson, Microsoft Research, UK
	Milad Shokouhi, Microsoft Research, UK
	Emine Yilmaz, Microsoft Research, UK

Sponsors

Microsoft Research
The Open University
British Computer Society - Information Retrieval Specialist Group (BCS-IRSG)
True Knowledge
Yahoo Research

Program Committee

Gianni Amati	FUB, Italy
Hany Azzam	Queen Mary University of London, UK
Richard Bache	University of Glasgow, UK
Jing Bai	Yahoo! Inc., USA
Mark Baillie	University of Strathclyde, UK
Roberto Basili	University of Rome Tor Vergata, Italy
Nick Belkin	Rutgers University, USA
Bodo Billerbeck	Microsoft Research, Cambridge, UK
Giorgio Brajnik	University of Udine, Italy
Peter Bruza	Queensland University of Technology, Australia
Di Cai	University of Glasgow, UK
Steven Cater	Kettering University, USA
Fabio Crestani	University of Lugano, Switzerland
Tamas Doszkocs	National Library of Medicine, USA
Hui Fang	University of Delaware, USA
Jan Frederik Forst	Queen Mary University of London, UK
Norbert Fuhr	University of Duisburg-Essen, Germany
Susan Gauch	University of Arkansas, USA
Thore Graepel	Microsoft Research, Cambridge, UK
Martin Halvey	University of Glasgow, UK
Claudia Hauff	University of Twente, The Netherlands
Ben He	University of Glasgow, UK
Djoerd Hiemstra	University of Twente, The Netherlands
Eduard Hoenkamp	Maastricht University, The Netherlands
Qiang Huang	The Robert Gordon University, UK
Jimmy Huang	York University, Canada
Theo Huibers	University of Twente, The Netherlands
Peter Ingwersen	Royal School of Library and Information Science, Denmark
Kalervo Jarvelin	Tampere University, Finland
Gareth Jones	Dublin City University, Ireland
April Kontostathis	Ursinus College, USA
Udo Kruschwitz	University of Essex, UK
Mounia Lalmas	University of Glasgow, UK
Wai Lam	Chinese University of Hong Kong, Hong Kong
Birger Larsen	Royal School of Library and Information Science, Denmark
Raymond Lau	City University of Hong Kong, Hong Kong
Victor Lavrenko	University of Edinburgh, UK
Christina Lioma	K.U. Leuven, Belgium

David Losada	Universidad de Santiago de Compostela, Spain
Robert Luk	Hong Kong Polytechnic University, Hong Kong
Massimo Melucci	University of Padua, Italy
Stefano Mizzaro	University of Udine, Italy
Dunja Mladenic	J. Stefan Institute, Slovenia
Nikolaos Nanas	Centre for Research and Technology - Thessaly, Greece
Jian-Yun Nie	University of Montreal, Canada
Paul Ogilvie	mSpoke, USA
Iadh Ounis	University of Glasgow, UK
Benjamin Piwowarski	University of Glasgow, UK
Filip Radlinski	Microsoft Research, Cambridge, UK
Vijay Raghavan	University of Louisiana at Lafayette, USA
Maarten de Rijke	University of Amsterdam, The Netherlands
Stephen Robertson	Microsoft Research, Cambridge, UK
Ian Ruthven	University of Strathclyde, UK
Milad Shokouhi	Microsoft Research, Cambridge, UK
Laurianne Sitbon	National ICT Centre, Australia
Edward Snelson	Microsoft Research, Cambridge, UK
Amanda Spink	Queensland University of Technology, Australia
Paul Thomas	CSIRO ICT Centre, Australia
Olga Vechtomova	University of Waterloo, Canada
Vishwa Vinay	Microsoft Research, Cambridge, UK
Jun Wang	The Robert Gordon University, UK
Jun Wang	University College London, UK
Wensi Xi	Google, USA
Emine Yilmaz	Microsoft Research, Cambridge, UK
Dell Zhang	Birkbeck, University of London, UK
Peng Zhang	The Robert Gordon University, UK
Jianhan Zhu	University College London, UK
Guido Zuccon	University of Glasgow, UK

Table of Contents

Invited Talk

Regular Papers

Efficiency

Retrieval Models

Query and Term Models

Evaluation

Novelty and Diversity

Short Papers

New Perspectives in IR

Retrieval Models

User Aspects

Web Models

Posters

Is There Something Quantum-Like about the Human Mental Lexicon?

Peter Bruza

Faculty of Science and Technology
Queensland University of Technology
p.bruza@qut.edu.au

Abstract. This talk proceeds from the premise that IR should engage in a more substantial dialogue with cognitive science. After all, how users decide relevance, or how they chose terms to modify a query are processes rooted in human cognition. Recently, there has been a growing literature applying quantum theory (QT) to model cognitive phenomena. This talk will survey recent research, in particular, modelling interference effects in human decision making. One aspect of QT will be illustrated - how quantum entanglement can be used to model word associations in human memory. The implications of this will be briefly discussed in terms of a new approach for modelling concept combinations. Tentative links to human abductive reasoning will also be drawn. The basic theme behind this talk is QT can potentially provide a new genre of information processing models (including search) more aligned with human cognition.

Acknowledgments. This research is supported in part by the Australian Research Council Discovery grant DP0773341.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, p. 1, 2009.

Probably Approximately Correct Search

Ingemar J. Cox[1], Ruoxun Fu[1], and Lars Kai Hansen[2]

[1] University College London
[2] Technical University of Denmark
ingemar@cs.ucl.ac.uk

Abstract. We consider the problem of searching a document collection using a set of independent computers. That is, the computers do *not* cooperate with one another either (i) to acquire their local index of documents or (ii) during the retrieval of a document. During the acquisition phase, each computer is assumed to randomly sample a subset of the entire collection. During retrieval, the query is issued to a random subset of computers, each of which returns its results to the query-issuer, who consolidates the results. We examine how the number of computers, and the fraction of the collection that each computer indexes, affects performance in comparison to a traditional deterministic configuration. We provide analytic formulae that, given the number of computers and the fraction of the collection each computer indexes, provide the probability of an approximately correct search, where a "correct search" is defined to be the result of a deterministic search on the entire collection. We show that the randomized distributed search algorithm can have acceptable performance under a range of parameters settings. Simulation results confirm our analysis.

1 Introduction

Searching the Web is critical to the Web's success. Search is now common - Americans alone are estimated to have performed over 13 billion searches in February 2009 [9]. And the size of the indexed web is now estimated to be about 65 billion webpages, of which Google is estimated to index over 17 billion pages [16].

The frequency of searches together with the size of the index prohibit a single computer being able to cope with the computational load. Consequently, a variety of computer architectures have been proposed. Commercial search engines such as Google, use an architecture where the the index is distributed (and arguably "virtually centralized") over a number of disjoint partitions [1]. And within each partition, the partial index is replicated across a number of machines. A query must be sent to one machine in each partition and their partial responses are then consolidated before being returned to the user. The number of partitions and the number of machines per partition is a careful balance between throughput and latency [6]. Changes to the collection or to the query distribution may necessitate that the index be repartitioned, a process than can be quite complex and time consuming [6]. Note that while the index is distributed across machines, the machines themselves are typically housed within a central server facility.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 2–16, 2009.

Peer-to-peer networks offer a more geographically dispersed arrangement of machines that are not centrally managed. This has the benefit of not requiring an expensive centralized server facility. However, the lack of a centralized management can complicate the communication process. And the storage and computational capabilities of peers may be much less than for nodes in a commercial search engine. Li *et al.* [5] provide an overview of the feasibility of peer-to-peer web indexing and search. Their analysis assumes a deterministic system in which, if necessary, a query is sent to all peers in the network, for example. The authors do comment on the possibility of "compromising result quality" by streaming the results to the users based on incremental intersection. However such a "compromise" is quite different from the non-deterministic search proposed here.

In this paper, we investigate the expected performance of a non-deterministic information retrieval system consisting of a set of independent computers. We define a non-deterministic information retrieval system to be one in which (a) the set of unique documents indexed may be selected (in part) randomly and/or (b) the response to a query may (in part) be a function of a random process. By "independent computers" we mean computers that do not communicate between one another for the purposes of either building the index, or responding to a query. The absence of communication/coordination between computers prevents the non-deterministic IR system from overloading the communication infrastructure, and provides a system architecture that is very scalable and reconfigurable.

Our system assumes two capabilities. First, the ability to randomly sample documents from a collection. And second, the ability to randomly sample/query computers within the network. The random sampling of documents within a collection, is, of course, trivial if the collection is available as a static document set with limited number of documents. However, if the collection is considered to be the Web, then it is necessary to randomly sample pages from the Web. This is more difficult. A comparison of several techniques can be found in [2,8]. The random sampling of computers within a network is straightforward when the computers are a part of a "centralized" cluster. And recent work [4,14,13], based on distributed hash tables, provides algorithms for choosing a random peer within a peer-to-peer network.

We are interested in comparing the performance of non-deterministic and deterministic IR systems. In this regard, we consider the results of the deterministic system to be correct, i.e. we are not judging the performance of our system based on standard IR metrics such a mean average precision (MAP). Rather, given a deterministic implementation of a specific IR system, how close will the outputs of a non-deterministic system be to the deterministic system? Given this measure, we refer to our system as a PAC (probably approximately correct) IR system, in (broad) analogy with PAC learning [15].

In Section 2 we first define a number of terms and concepts before deriving analytic expressions describing the expected coverage of a PAC IR system and its expected level of correctness. Section 3 then discusses the performance of a PAC for two specific configurations, the first of which models the architecture used by search engine services, and the second models a hypothetical peer-to-peer network configuration. Section 4 provides simulation results that support the previous theoretical analysis. Finally, Section 5 summarizes our results and suggests avenues for future work.

2 Framework

Our model of an IR system assumes a set of computers, and that each independently samples a fraction of available documents to construct a corresponding inverted index. We refer to this as the acquisition stage. Next, a user query is issued to a (small) subset of these computers and each computer independently responds with a corresponding result set. These result sets are then merged by the query issuer to produce the overal result set. We refer to this as the retrieval stage.

In the next Section, Section 2.1, we first define a variety of terms and concepts. Section 2.2 then considers the acquisition stage, and derives an analytic model for the expected coverage of our PAC IR system. This model is then used in Section 2.3 to derive an analytic model for the correctness of a PAC IR system.

2.1 Definitions

The entire set of unique documents is referred to as the *collection*. The size of the collection is denoted by N. For the Web, N ranges from 17 to 65 billion webpages, as noted earlier.

Let K represent the total number of computers available to perform searches. Note that in the case of peer-to-peer networks, K is not constant. However, in such a case, let us assume K represents the average number of available computers. For simplicity, we assume that the computers are homogeneous. However, this is not needed in practice.

Each computer is assumed to be capable of indexing n unique documents, which form an individual sample from the collection. We assume that $n \leq N$, and, in practice, normally $n \ll N$. We define the "collection sample" as the union of individual samples. As such, the collection sample may well contain duplicate documents. We define coverage as the ratio of the number of unique documents in a collection sample to the size of the collection. Finally, during retrieval, we query a subset, k', of computers, and the union of their indices is known as the "retrieval index".

2.2 Sampling the Collection

In order to index the N distinct documents, the K computers must sample the collection (Web). In our analysis we assume that each computer operates independently, with no cooperation between computers. In such a scenario, there is no guarantee that the samples on each computer will be disjoint. In fact, it is almost certain that documents will be sampled more than once, i.e. they will be indexed by more than one computer. This redundancy is, in fact, helpful. First, it provides tolerance to node failures, and to the dynamic entry and exit of nodes in a peer-to-peer network. Second, the redundancy allows only a subset of nodes to answer a query (see Section 2.3), which both reduces the communication overhead and increases the throughput, i.e. the number of queries that can be answered per second.

Independent sampling of the N documents in the collection by each of the K computers is analogous to having an urn with N labeled balls. Each of K people, then randomly choose n balls each. An individual chooses his/her n balls without replacement, thereby guaranteeing that there is no repetition on a single computer. After indexing the n balls, they are returned to the urn. Thus, the next person may also randomly select balls that have been previously chosen by other people (i.e. indexed by other computers).

The key question to answer is, how many different balls have been drawn from the urn after all K people have each randomly picked n balls? The answer to this question determines the coverage obtained after all K computers have sampled n documents.

Obviously, the coverage ranges from $\frac{n}{N}$ in the worse case, where all computers sample the same set of n documents, to $\frac{\min(N,Kn)}{N}$ in the best case, where each computer's sample is disjoint from all other computers' samples. Treating the coverage as a random variable, we need to understand its probability distribution. Of course the complete probability distribution may be quite complicated. However, from a practical point of view, we believe that an analysis on the expected coverage would be sufficient to explain the underlining rules of our algorithm.

The probability of a ball being picked by a single individual is $\frac{n}{N}$, and the probability of *not* being picked is therefore $1 - \frac{n}{N}$. Thus the probability, $P(\bar{d_i})$, that document d_i will *not* be picked by *any* of the K people is

$$P(\bar{d_i}) = \left(1 - \frac{n}{N}\right)^K \tag{1}$$

Thus, the probability, $P(d_i)$, of being chosen one or more times in the total sample is

$$P(d_i) = 1 - P(\bar{d_i}) = 1 - \left(1 - \frac{n}{N}\right)^K \tag{2}$$

and the expected number of distinct documents in our total sample, $\hat{N}$, is

$$\hat{N} = P(d_i)N = \left(1 - \left(1 - \frac{n}{N}\right)^K\right) N \tag{3}$$

To simplify our further analysis, let us now set

$$\epsilon = P(\bar{d_i}) = \left(1 - \frac{n}{N}\right)^K \tag{4}$$

Then

$$\hat{N} = N(1 - \epsilon). \tag{5}$$

And the expected coverage can be defined as

$$E(Coverage) = \frac{\hat{N}}{N} = 1 - \epsilon. \tag{6}$$

Thus, as ϵ approaches zero, our coverage approaches unity, i.e. we approach a complete sampling of the document collection.

We can use Equation (4) - Equation (6) to determine the coverage. We are interested in the relationship between coverage, the size of the individual sample, n, and the number of computers, K, given a certain collection size, N. In particular, given a collection, how many machines do we need, and what capacity should each of them have, to meet a desired level of performance for our system.

To help our analysis, let us first denote c as the size of the collection sample, where $c = Kn$. The collection sample, c, is treated as a constant in the following analysis. Also to simplify our analysis, let us first assume that $c \leq N$. From Equation (4), we have

$$\epsilon = \left(1 - \frac{n}{N}\right)^K = \left(1 + \frac{-\frac{c}{N}}{K}\right)^K \tag{7}$$

Thus, if the collection sample, $c = nK$ is a constant, then ϵ is a monotonically increasing function with respect to K. The smallest value of $\epsilon = (1 - \frac{n}{N})$ occurs when $K = 1$. In this case, n is at its largest, and the coverage is maximized since there are no duplicates in our collection sample. Conversely, as the number of computers, K, increases, ϵ increases, approaching the limit of $e^{-\frac{c}{N}}$ as K approaches infinity. The proof is shown as below.

From the property of exponential functions, we know that

$$e^x = \lim_{n \to \infty} \left(1 + \frac{x}{n}\right)^n \tag{8}$$

From Equations (7) and (8), we have that

$$\lim_{K \to \infty} \epsilon = \lim_{K \to \infty} \left(1 + \frac{-\frac{c}{N}}{K}\right)^K = e^{-\frac{c}{N}} \tag{9}$$

Next, the derivative of ϵ with respect to K is

$$\frac{\partial \epsilon}{\partial K} = \epsilon \left(\ln\left(1 + \left(-\frac{c}{NK}\right)\right) - \left(\frac{-\frac{c}{NK}}{1 + (-\frac{c}{NK})}\right)\right)$$

From the property of natural logarithms, we also have

$$\ln(1 + h) \geq (\frac{h}{1 + h}), \text{for } h \geq -1 \tag{10}$$

Since

$$n \leq N \Rightarrow nK \leq NK \Rightarrow -\frac{c}{NK} = -\frac{nK}{NK} \geq -1$$

So $\ln\left(1 + (-\frac{c}{NK})\right) - \left(\frac{-\frac{c}{NK}}{1+(-\frac{c}{NK})}\right) \geq 0$, because $\epsilon \geq 0$, we have

$$\frac{\partial \epsilon}{\partial K} = \epsilon \left(\ln\left(1 + (-\frac{c}{NK})\right) - \left(\frac{-\frac{c}{NK}}{1 + (-\frac{c}{NK})}\right)\right) \geq 0 \qquad (11)$$

Combining Equations (9) and (11), we show that the ϵ increases monotonically with an upper bound of $e^{-\frac{c}{N}}$, as K increases. Thus, the expected coverage ranges from $(1 - e^{-\frac{c}{N}}$ to $\frac{c}{N}]$. Figure 1 plots ϵ and coverage for the case where $N = 1000000$ and $c = N$.

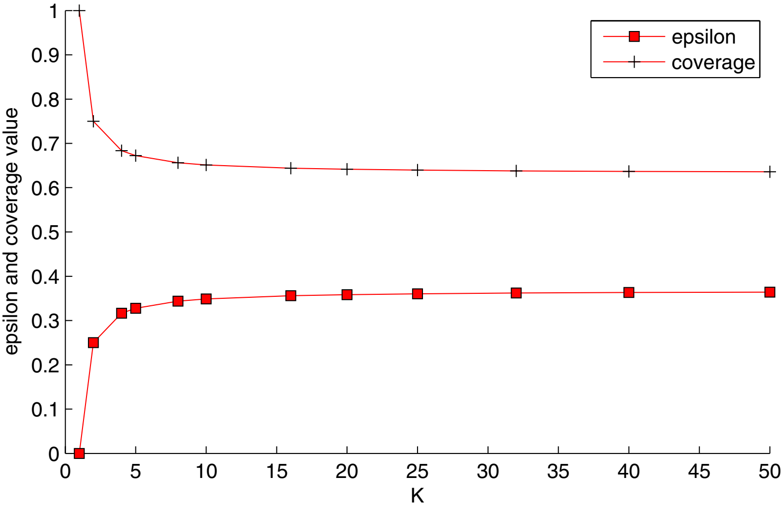


Fig. 1. Simulation calculating ϵ and coverage as a function of the number of computers, K, when the number of computers, $N = 1000000$, and the collection sample is $c = N$

This monotonic property remains true when we relax the assumption that $c \leq N$, and allow $c > N$, provided $n \leq N$. In this case, K cannot start from 1 since it would imply that $n > N$. Let us define K_{min} as the smallest value of K such that the property $n \leq N$ is maintained. Then, a more general form of coverage can be written as $(1 - e^{-\frac{c}{N}}, 1 - (1 - \frac{c}{NK_{min}})^{K_{min}}]$.

In summary, for any given c, we have a lower bound, $1 - e^{-\frac{c}{N}}$, for the expected coverage. The smallest coverage occurs when $K = c$ and $n = 1$, and approaches $1 - e^{-\frac{c}{N}}$ if K is large enough. Conversely, coverage is maximized when $K = K_{min}$, and is given by $1 - (1 - \frac{c}{NK_{min}})^{K_{min}}$.

Unfortunately, coverage is not our only concern. We must also consider the throughput of the system, as well as the system's latency. However though smaller K promises a larger coverage, it results in a larger individual sample, n.

Let us define k' as the number of machines that process a query simultaneously, and p as the number of documents that each machine can process in a unit time. Then, the query rate, T, can be defined as

$$T = \frac{K}{k'} \times \frac{p}{n} \tag{12}$$

The first factor represents the query throughput of the system, and the second factor is the inverse of the latency. Suppose k' and the collection sample size, c, are fixed, then

$$T = \frac{K^2 \times p}{k' \times c} \propto K^2 \tag{13}$$

Obviously larger K increases the query throughput, but, as we discussed earlier, a larger K decreases coverage, when our sample collection size, c, is fixed. Thus, for a given c, choosing appropriate values of K and k' is a tradeoff between coverage and query rate, and will depend on the application.

For example, consider the case where the size of collection sample, c, is equal to the size of the collection, N. Using Equation 9, we can easily infer that ϵ tends to be $e^{-1} = 0.367$ with a large K. Inserting this value in Equation (6), we see that when we sample N documents, our expected coverage is at least $1 - \epsilon = 0.63$.

Next, let us assume we want complete coverage. Of course, this cannot be guaranteed, but we can set ϵ to a small value such that the probability of missing a document is low. For example, consider the case when $\epsilon = 10^{-2}$, say. That is, 99% coverage. In this case, from Equation (9) we have

$$\epsilon = e^{-\frac{c}{N}} = 10^{-2} \Rightarrow \frac{c}{N} \approx 4.6$$

Thus, if the size of the collection sample is 4.6 times the size of collection, we can expect 99% coverage of the collection.

2.3 Retrieval

The previous theoretical analysis elucidated the connection between (i) the number of computers, K, (ii) the size of each computer's sample, n and (iii) the fraction of the collection that is not indexed, ϵ. By increasing K and/or n, we can make ϵ as small as desired. Of course, in practice, economic considerations can limit the values of both K and n.

When performing retrieval within such an architecture, we wish to send the query to k' randomly chosen nodes, where $k' \leq K$, and normally $k' \ll K$. This is because it is necessary to (i) limit the amount of communication generated by a query, (ii) limit the computational resources expended in responding to a query, and (iii) limit the latency between query issue and response.

Clearly, if we only interrogate k' machines, we cannot guarantee the coverage provided by all K machines. However, Equation (3) can be used to determine the expected size of the index used during retrieval, i.e. the expected number of distinct documents in the retrieval index. For this, we simply have to replace K with k'.

$$\hat{N}' = P(d_i')N = \left(1 - \left(1 - \frac{n}{N}\right)^{k'}\right) N \tag{14}$$

The probability of any document being present in the retrieval index is then

$$P(d_i') = 1 - \left(1 - \frac{n}{N}\right)^{k'} \tag{15}$$

In practice, information retrieval systems are seldom evaluated based on a single target document. Instead, performance metrics such as precision and recall are often used. In our case, we assume that the retrieval model is identical, irrespective of whether we are using a deterministic or non-deterministic search architecture. Thus, if we want to compare our PAC strategy to a deterministic implementation of the IR system, we need to consider what the expected overlap in the two result sets is. Thus, given the top-r documents from the deterministic system, what is the probability that our PAC IR system will retrieve r' documents from r, where $r' \leq r$.

We know from the Equation (15) that the probability of a specific document being present in the result set is $P(d_i')$. Thus, the probability of exactly r' documents from r being present in the result set is

$$P(r') = \binom{r}{r'} P(d_i')^{r'}(1 - P(d_i'))^{r-r'} \tag{16}$$

This is a standard binomial distribution, and the expectation of r' is therefore

$$E(r') = rP(d_i') \tag{17}$$

Equation (17) indicates that acceptable performance using PAC search can be achieved provided the probability, $P(d_i')$, is sufficiently high. We will discuss this problem in more detail in the next section, where we consider two practical applications.

3 Discussion

Information retrieval systems can be broadly categorized into one of three architectures, namely (i) single server search, (ii) distributed and, arguably, “virtually centralized” search, and (iii) peer-to-peer decentralized search. We do not consider the first case, as we assume that our collection and/or query rate is too large to be handled by a single machine. The second case represents the architecture used by commercial search engines such as Google [1]. Finally a variety of peer-to-peer decentralized architectures have been proposed and deployed [11,12,3,7,18,17,10] with a variety of search capabilities. In the following two subsections we examine the second case and the third case, respectively.

3.1 Distributed Search

Due to the rate of queries and the huge size of the Web, modern commercial search engines partition the Web index over many machines. A response to a query requires each partition to be independently searched. Each partition (Google refers to them as index shards [1]) contains a "randomly chosen subset of documents from the full index" [1]. Note, however, that while the documents may be chosen at random, each partition is disjoint. In addition, replicas are added to each partition to increase the query throughput. This architecture is referred to as a distributed cluster architecture.

The key parameter of such an architecture is the tradeoff between the replication and partitioning. While increasing the partitioning level, which reduce the replication level, improves the query completion time since more machines process the same query simultaneously, the reduced replication level decreases the number of different queries that can be answered at the same time. A crucial problem faced by these engines is to find a best compromise between partitioning and replication, especially as the data set and the query rate change continuously. Clearly this compromise changes over time, as the database and query loads change. However, changing the partitions and replications can be expensive in both time and bandwidth, as reported in [6].

In [6] it is claimed that Google partitions its index into 1000 disjoint sets. Thus, the number of documents indexed by a single machine is $\frac{n}{N} = \frac{1}{1000}$. It is further claimed that the data in any partition is replicated over 300 machines, so the total number of machines is $K = 300,000$, and the total number of samples is $Kn = 300N$. Let us now examine the performance of such a system, when configured for PAC IR.

First, let us consider the expected coverage when each of the K machines, independently samples 0.1% of the Web. Solving for ϵ in Equations (7) and (9), we have

$$\epsilon = \left(1 + \left(\frac{-300}{300000}\right)\right)^{300000} \approx e^{-300}$$

This is a very small number and indicates that if all 300,000 machines were to each, independently randomly sample and index 0.1% of the Web, then it is almost certain the every document on the Web would be contained in the combined index.

For the query part, let us consider the configuration ascribed to Google, in which 1000 machines, one per partition, are used to service each query. In this case, $k' = 1000$ and $\frac{n}{N} = \frac{1}{1000}$, as before. Substituting in Equation (15) we get

$$P(d_i') = 1 - \left(1 - \frac{1}{1000}\right)^{1000} \approx 0.63$$

Thus, if a user is looking for a particular document, there is a 63% chance that it is present in a subset of 1000 randomly chosen nodes. That is, approximately two thirds of the time, the user will find the specific target document.

Table 1. The probability of exactly r' documents being present in the top-10

r'	$P_r(d_1 \cdots d_{r'})$
0	0.000045173
1	0.00077682
2	0.0060113
3	0.027566
4	0.082957
5	0.17119
6	0.24532
7	0.24106
8	0.15545
9	0.059405
10	0.010216

Assuming we are primarily interested in the top-10 results, i.e. $r = 10$, and given $P(d_i') = 0.63$, substituting in Equation (17), gives

$$E(r') = 10 \times 0.63 = 6.3$$

This shows that we can, on average, expect 6 documents from the top-10 retrieved by a deterministic search algorithm to be present in our PAC IR top-10.

We can also use Equation (16) to calculate the probabilities for all possible r'. These probabilities are enumerated in Table 1 for $r = 10$ and $0 \leq r' \leq 10$.

Table 1 indicates that there is over an 88% chance of retrieving 5 or more documents in common with the deterministic solution. And the most likely situation, occurring about 25% of the time, is that 6 out of the 10 documents will be common. There is approximately a 1% chance that the PAC search result set will be identical to the deterministic case. In contrast, the likelihood that the PAC search results do not contain any of the documents from the deterministic case, occurs less than 0.01% of the time.

In summary, the performance of our PAC IR system is approximately 63% of the deterministic system, when utilizing equivalent resources. Of course, we can improve performance by simply increasing the number of machines the query is sent to. For example, if we send the query to 2000 servers, then the query correctness increases to 86%. Unfortunately, this is at the expense of halving the query throughput. However, this example serves to highlight the flexibility of PAC search, which allows accuracy to be traded for throughput. That is, a PAC IR system could choose to tradeoff accuracy for query throughput during peak load periods.

Due to the unstructured nature of the PAC IR system, it is also straighforward to add and remove machines as well as adjust the data present on a machine.

3.2 Peer-to-Peer Decentralized Search

Another possible implementation of PAC IR is in peer-to-peer decentralized search. Following the estimation data in [5], suppose we have a peer-to-peer network with one million machines ($K = 10^6$). Let us further assume that every machine can provide 1GB for storing the index. If every document has, on average, 1000 distinct terms, and each term posting requires 20 bytes, then every document consumes 20k bytes in the index, and each machine can therefore index 50k documents ($n = 5 \times 10^4$). Thus the whole network has $Kn = 5 \times 10^{10}$ documents.

Now let us consider the case where we wish the peer-to-peer PAC IR system to index 17 billion documents, which is the same of the estimated size of Google's collection. Thus, the coverage obtained by the collection sampling is

$$E(Coverage) = 1 - \left(1 - \frac{5 \times 10^4}{1.7 \times 10^{10}}\right)^{10^6} = 0.947$$

This is not particularly surprising given that the size of our collection sample, Kn, is about 3 times the collection size.

During retrieval we must once again transmit the query to only a subset of the 1 million machines. If we assume that the query is sent to 10000 machines [5], then $k' = 10000$, and we have

$$P(d_i') = 1 - \left(1 - \frac{5 \times 10^4}{1.7 \times 10^{10}}\right)^{10000} = 0.03$$

The expected number of documents common to the top-10 generated by a deterministic search is then $E(r') = 10 \times 0.03 = 0.3$, i.e. on average, there is less than one document in common.

It is tempting to assume that this poor performance is due to the random nature of PAC IR. However, if we consider the expected number of distinct documents in a random selection of 10000 machines, we have

$$n_{distinct} = \left(1 - \left(1 - \frac{5 \times 10^4}{1.7 \times 10^{10}}\right)^{10000}\right) \times (1.7 \times 10^{10}) \approx 4.93 \times 10^8$$

In comparison, if each machine sample is disjoint from one another, we have $5 \times 10^4 \times 10^4 = 5 \times 10^8$ distinct documents. Thus, the coverage provided by the random sampling is $\frac{4.99}{5} = 98.6\%$ of the best possible coverage.

In fact, the root cause of the poor performance is due to the low capacity of each machine. If we wish to reach a PAC performance of 63%, we need to query $340,000$ machines.

This conclusion can also be reached in the Bubble Storm algorithm[13], in which the probability of a query meeting a document is $1 - e^{-\frac{k'g}{K}}$, where k' is the query replication number, g is the document replication number and K is the total number of machines in the network. Following the estimation data above, the average document replication number is $g = \frac{Kn}{N} \approx 3$, $k' = 10000$ and $K = 10^6$. So the probability of a query meeting a document is $1 - e^{-\frac{3 \times 10000}{10^6}} = 0.03$, which is similar to the result of our PAC IR algorithm. And, of course, too low to be practical.

The simple solution to the problem is to increase each machine's capacity. Suppose each machine can provide 340GB for storing data, then

$$P(d_i') = 1 - \left(1 - \frac{5 \times 10^4 \times 340}{1.7 \times 10^{10}}\right)^{1000} = 0.63$$

Thus $E(r') = 10 \times 0.63 = 6.3$. However, it seems unlikely that most peers can provide this storage requirement.

4 Simulation

The previous theoretical analysis examined the *expected* coverage and corresponding query performance, which is the result of averaging over many trials. In practice, any configuration for a PAC IR system represents a single instance or trail. Thus, it is interesting to investigate the standard deviation form the expected value, across trails. Clearly, we would like this to be small.

We investigated this issue using a simulation with different settings of machine capacity (n), number of machines (K) and collection size (N). In the first simulation, we manually generated a collection with $1e + 6$ documents ($N = 1e + 6$), and set $n = 1000, K = 1000$. This synthetic collection was simply a set of document identifiers. In each trial, each of the K machines samples n documents to form a collection sample, which is then stored in disk. Then we repeat this process to generate 20 trails and a corresponding 20 collection samples. Next, we randomly generated 100 test queries and computed the top-10 ranking results from the original full collection. Then, for each trial, the queries are replicated to all 1000 nodes and an averaged query performance on each collection sample is calculated. The results for each trial were then avaerged to provide an estimate of the expected values for coverage and query performance.

In the second simulation, we change K to be 2000 with all other parameters being the same as the first one.

In the third simulation, we use TREC45 as our experiment environment to test the performance of PAC. TREC45 contains about 550,000 documents, i.e. $N = 556079$. All other settings are set the same as for the first simulation.

The results from Tables 2 and 3 show that the variation across trials is very small. This is very encouraging and supports our analysis in section 2, which indicates that in spite of the random nature of the PAC search, the most common outcomes for coverage and query performance concentrate in a short range centered around their expectations.

Table 2. Comparison of expected coverage, average coverage and standard deviation across 20 trials

Simulation	Expectation	Average	Std dev.
1	0.6323	0.6322	0.0003
2	0.8648	0.8648	0.0004
3	0.8347	0.8346	0.0004

Table 3. Comparison of expected query performance, average query performance and standard deviation across 20 trials

Simulation	Expectation	Average	Std dev.
1	0.6323	0.6264	0.0135
2	0.8648	0.8636	0.0124
3	0.8347	0.8377	0.007

5 Conclusion

We examined the problem of non-deterministic search in which a set of computers (i) independently sample the collection/Web and (ii) queries are sent to a random subset of computers. Equations are derived for the expected coverage of the sample collection, and the accuracy of the retrieval results. The latter is measured with respect to the results provided by a deterministic IR system. Under the assumption that the deterministic system provides correct result, we consider the probability of being approximately correct. We therefore describe our approach as PAC search.

Our analysis of PAC IR in the context of commercial search engines suggest that a performance level of 63% can be achieved using the same amount of storage and computation. However, while the performance is lower, we believe that the PAC IR architecture may be simpler to manage. Moreover, more sophisticated implementations might close this performance gap.

PAC IR was also analyzed in the context of peer-to-peer decentralized web search. The key problem with such a configuration appears to be the much small storage available on any machine. Consequently, it would be necessary to send the query to many more computers, and the communication overhead may then be too high.

The fact that a query is sent to a random set of machines means that the same search, issued multiple time, is likely to produce different results. Users may find this disconcerting. However, if a pseudo-random set of machines is selected based on a function (hash) of the query, then the result set would remain the same each time the same query is issued. For common queries, additional random machines could be queried to determine if better results exist within the sample collection. If so, these documents could be indexed by the pseudo-random set of machines corresponding to the query. More generally, for common queries, it is interesting to consider how to optimally learn the best set of k machines to answer the query.

A further level of optimization is the caching of query results. First, it would be interesting to analyze the expected cache hit rate for a given distribution of queries when a query is sent to a random set of machines. And a similar analysis should be performed when the query is sent to a pseudo-random (deterministic) set of nodes.

A key assumption in our analysis is the ability to randomly sample the collection. This is difficult, but certainly possible. Moreover, in the case of a centrally managed system, common to commercial search engines, it would not be necessary to for each machine to independently sample the Web. Rather, a centralized crawler could still be used, and the documents from this crawl could be randomly (and non-disjointly) partitioned across the computers.

We have also implicitly assumed that the deterministic and non-deterministic IR systems both implement the same underlying retrieval model. Usually, most retrieval models have parameter values that are based on the statistics of the collection. However, for the PAC IR system, each computer only has access to its local sample. Future work is needed to determine if, and under what conditions, the statistics of the local samples will be sufficiently close to the statistics of the overall collection.

Acknowledgements

The authors thank Vishwa Vinay of Microsoft Research, Cambridge, and Brad Karp, David Rosenblum and Jun Wang of UCL for useful discussion on earlier drafts of this paper.

References

1. Barroso, L.A., Dean, J., Holzle, U.: Web search for a planet: The google cluster architecture. IEEE Micro. 23(2), 22–28 (2003)
2. Baykan, E., de Castelberg, S., Henzinger, M.: A comparison of techniques for sampling web pages. In: Dagstuhl Seminar Proceedings, vol. 09001. Schloss Dagstuhl, Germany (2009)
3. Harren, M., Hellerstein, J.M., Huebsch, R., Loo, B.T., Shenker, S., Stoica, I.: Complex queries in dht-based peer-to-peer networks. In: Druschel, P., Kaashoek, M.F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, p. 242. Springer, Heidelberg (2002)
4. King, V., Saia, J.: Choosing a random peer. In: PODC, pp. 125–130 (2004)
5. Li, J., Loo, B.T., Hellerstein, J.M., Kaashoek, M.F., Krager, D.R., Morris, R.: On the feasibility of peer-to-peer web indexing and search. In: Kaashoek, M.F., Stoica, I. (eds.) IPTPS 2003. LNCS, vol. 2735, pp. 207–215. Springer, Heidelberg (2003)
6. Raiciu, C., Huici, F., Handley, M., Rosenblum, D.: ROAR: Increasing the flexibility and performance of distributed search. In: Proc. ACM SIGCOMM 2009 Conference on Data Communication, SIGCOMM 2009 (2009)
7. Reynolds, P., Vahdat, A.: Efficient peer-to-peer keyword searching. In: Proceedings of the International Middleware Conference (2003)
8. Rusmevichientong, P., Pennock, D.M., Lawrence, S., Giles, C.L.: Methods for sampling pages uniformly from the world wide web. In: Proc. AAAI Fall Symposium on Using Uncertainty Within Computation, pp. 121–128 (2001)
9. `http://news.ebrandz.com/google/2009/2495-google-continues-to-lead-february-2009-us-search-engine-rankings-comscore-.html` (2009)
10. Skobeltsyn, G., Luu, T., Zarko, I.P., Rajman, M., Aberer, K.: Web text retrieval with a p2p query-driven index. In: SIGIR, pp. 679–686 (2007)
11. Stoica, I., Morris, R., karger, D., Kaashoek, F., Balakrishnan, H.: Chord: Scalable peer-to-peer lookup service for internet applications. In: Proceedings of the 2001 ACM SIGCOMM Conference, pp. 149–160 (2001)

12. Tang, C., Xu, Z., Mahalingam, M.: psearch: Information retrieval in structured overlays. In: HotNets-I (2002)
13. Terpstra, W.W., kangasharju, J., Leng, C., Buchmann, A.P.: Bubblestorm: resilient, probabilistic, and exhaustive peer-to-peer search. In: SIGGCOMM 2007 (2007)
14. Terpstra, W.W., Leng, C., Buchmann, A.P.: Bubblestorm: Analysis of probabilistic exhaustive search in a heterogeneous peer-to-peer system. In: Technical Report TUD-CS-2007-2 (2007)
15. Valiant, L.G.: A theory of the learnable. Communications of the ACM 27(11), 1134–1142 (1984)
16. http://www.worldwidewebsize.com/ (2009)
17. Yang, K.-H., Ho, J.-M.: Proof: A dht-based peer-to-peer search engine. In: Conference on Web Intelligence, pp. 702–708 (2006)
18. Yang, Y., Dunlap, R., Rexroad, M., Cooper, B.F.: Performance of full text search in structured and unstructured peer-to-peer systems. In: INFOCOM (2006)

PageRank: Splitting Homogeneous Singular Linear Systems of Index One

Douglas V. de Jager and Jeremy T. Bradley

Department of Computing, Imperial College London,
180 Queen's Gate, London SW7 2BZ, United Kingdom

Abstract. The PageRank algorithm is used today within web information retrieval to provide a content-neutral ranking metric over web pages. It employs power method iterations to solve for the steady-state vector of a DTMC. The defining one-step probability transition matrix of this DTMC is derived from the hyperlink structure of the web and a model of web surfing behaviour which accounts for user bookmarks and memorised URLs.

In this paper we look to provide a more accessible, more broadly applicable explanation than has been given in the literature of how to make PageRank calculation more tractable through removal of the dangling-page matrix. This allows web pages without outgoing links to be removed before we employ power method iterations. It also allows decomposition of the problem according to irreducible subcomponents of the original transition matrix. Our explanation also covers a PageRank extension to accommodate TrustRank. In setting out our alternative explanation, we introduce and apply a general linear algebraic theorem which allows us to map homogeneous singular linear systems of index one to inhomogeneous non-singular linear systems with a shared solution vector. As an aside, we show in this paper that irreducibility is not required for PageRank to be well-defined.

1 Introduction

The PageRank metric is a widely-used hyperlink-based estimate of the relative importance of web pages [1,2]. The standard algorithm for determining PageRank uses power method iterations to solve for the steady-state vector of a DTMC. The one-step probability transition matrix which defines the DTMC is derived from a web graph that reflects the hyperlink structure of the web and a user-centred model of web-surfing behaviour. The model provides a mathematical account of how users make use of web bookmarks and also of what they do when at web pages without any outgoing links.

In July of 2008, it was announced by Google that the company's crawlers had found more than one trillion current unique web pages, and that the web was growing at several billion new pages every day [3]. This is in contrast with Google's index of 26 million web pages in 1998 [3]. Despite its size, Google's index is only a fraction of the total number of indexable web pages in existence. This is

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 17–28, 2009.

because many sites are currently difficult to index—because of technologies like JavaScript and Flash, and also because certain sites require appropriate handling of forms, drop-downs, and so on. Also, by design, certain sites prevent access by Google's crawlers to many of their constituent web pages through robots.txt and nofollow links.

In [4] an investigation was presented into this rapid growth of the web. It was argued that the recent acceleration of growth has been driven in particular by a growing percentage of web pages without outgoing links—upward of fifty percent—and an analysis was provided into the different sorts of web pages which are classed by search engines as having no outgoing links.

In this paper we show how to reduce the complexity of PageRank calculation by partitioning the treatment of web pages with and without outgoing links, such that only pages with outgoing links are required during the power method iterations. As an added benefit, this approach also permits decomposition of the PageRank problem according to connected subcomponents of the original transition matrix [5]. We show this by presenting PageRank as a special case of a broader class of problem. Our proposal is an alternative formulation of linear algebraic proposals made in [6,7]—which are extensions of a lumping proposal in [8]. In this paper we also consider a PageRank extension considered via lumpability theory in [9,10], which allows for TrustRank [11]. We show that this extension is also a special case of the same broader class of problem, and that it can be handled similarly, in linear algebraic fashion. We suggest treating these proposals as companions to work in two other key research areas, research which focuses specifically on the size of the PageRank problem. The two research areas are asynchronous solution methods (where problem size requires the use of heterogeneous computing clusters) [12] and partitioning techniques for the PageRank problem across multiple processors [13,14,5].

The kernel of our proposal is a linear algebraic theorem which allows homogeneous singular linear systems of index one to be mapped to inhomogeneous nonsingular linear systems with a shared solution vector. This theorem is a general one. Its formulation was inspired by the PageRank equation. However, the theorem is not restricted in applicability to PageRank; nor indeed to DTMCs. The theorem may be used to apply novel solution methods to eigenvector problems (for example, asynchronous methods which require the spectral radius of the modulus equivalent of the coefficient matrix to be strictly less than one [15]), or it may be used to improve sparsity patterns or conditioning when using traditional solution methods to such problems.

As an aside to the core proposal, we show that, contrary to the standard presentation of PageRank in the literature, irreducibility is not required for PageRank to be well-defined. We show, in particular, that the personalisation vector needs not to be completely dense.

The paper is organised as follows. In Section 2 we review the conceptual model for PageRank. We set out what is now regarded as the standard definition of PageRank, and we present its sparse formulation. In Section 3 we extend the standard PageRank definition to allow for non-dense personalisation vectors.

In Section 4 we consider web pages without outgoing links. We introduce a theorem which allows us to map homogeneous singular linear systems of index one to inhomogeneous non-singular linear systems with a shared solution vector. Using this theorem we show how to employ the original transition matrix as a coefficient matrix when solving for standard PageRank—without an adjustment to deal with pages without outgoing links. We then extend our approach to deal with a generalisation of the PageRank definition which accounts for TrustRank.

2 Standard PageRank Definition

PageRank computation for the ranking of hypertext-linked web pages was originally outlined by Page and Brin [1,2]. Their approach was subsequently amended by Kamvar *et al.* [16]. This alternative formulation of PageRank and its computation is now generally regarded as providing the standard PageRank definition [17,18].

The standard conceptual model of PageRank is called the *random surfer* model. Consider a surfer who starts at a web page and picks one of the links on that page at random. On loading the next page, this process is repeated. If a *dangling page* (that is, a page without outgoing links—also referred to as a *cul de sac page*) is encountered, then the surfer chooses to visit a random page (as though going to a memorised link, or a bookmarked link). During normal browsing, the user may also decide, with a fixed probability, not to choose a link from the current page, but instead to jump at random to another page. In the latter case, to support both unbiased and personalised surfing behaviour, the model allows for the specification of a probability distribution of target pages.

The PageRank of a page is considered to be the limiting (steady-state) probability that the surfer is visiting a particular page after a large enough number of click-throughs. Calculating this probability vector corresponds to finding a dominant eigenvector of the modified web-graph transition matrix.

2.1 Random Surfer Model

In the random surfer model, the web is represented by a graph $G = (V, E)$, with web pages as the vertices, V, and the links between web pages as the edges, E. If a link exists from page u to page v then $(u \to v) \in E$.

To represent the following of hyperlinks, we construct a transition matrix P from the web graph, setting:

$$P_{ij} = \begin{cases} \frac{1}{\deg(u_i)} & : \text{ if } (u_i \to u_j) \in E \\ 0 & : \text{ otherwise} \end{cases} \qquad (1)$$

where $\deg(u)$ is the out-degree of vertex u, i.e. the number of outbound links from page u. From this definition, we see that if a page has no out-links, then this corresponds to a zero row in the matrix P. To represent the surfer's jumping from dangling pages, we construct a second matrix $D = \boldsymbol{d}\boldsymbol{p}^T$, which we refer to as the *dangling-page* matrix, where $\boldsymbol{d}$ and $\boldsymbol{p}$ are both column vectors, and

$$d_i = \begin{cases} 1: & \text{if } \deg(u_i) = 0 \\ 0: & \text{otherwise} \end{cases} \tag{2}$$

and $\boldsymbol{p}$ is the personalisation vector representing the probability distribution of destination pages when a random jump is made. Typically this distribution is taken to be uniform, i.e. $p_i = \frac{1}{n}$ for an n-page graph ($1 \leq i \leq n$). However, it need not be, as many distinct personalisation vectors may be used to represent different classes of user with different web browsing patterns. This flexibility comes at a cost, though, as each distinct personalisation vector requires an additional PageRank calculation.

Putting together the surfer's following of hyperlinks and their random jumping from dangling pages yields the stochastic matrix $P' = P + D$, where P' is a one-step probability transition matrix of a DTMC.

To represent the surfer's decision not to follow any of the current page links, but to jump instead to a random web page, we construct a *teleportation* matrix E, where $E_{ij} = p_j$ for all i, i.e. this random jump is also dictated by the personalisation vector.

Incorporating this matrix into the model gives:

$$A = cP' + (1 - c)E \tag{3}$$

where $0 < c < 1$, and c represents the probability that the user chooses to follow one of the links on the current pages—i.e. there is a probability of $(1-c)$ that the surfer randomly jumps to another page instead of following links on the current page.

This definition of A avoids two potential problems. The first is that P', although a valid DTMC transition matrix, is not necessarily irreducible and aperiodic. Taken together, these are a sufficient condition for the existence of a unique steady-state distribution [16,18]. Now, provided $p_i > 0$ for all $1 \leq i \leq n$, irreducibility and aperiodicity are trivially guaranteed.

The second problem relates to the rate of convergence of power method iterations used to compute the steady-state distribution. This rate depends on the reciprocal of the modulus of the subdominant eigenvalue (λ_2). For a general P', $|\lambda_2|$ may be very close to 1, resulting in a very poor rate of convergence. However, it has been shown that in the case of matrix A, $|\lambda_2| \leq c$, thus guaranteeing a good rate of convergence for the widely taken value of $c = 0.85$ [19].

Given the matrix A, we can now define the unique PageRank vector, $\boldsymbol{\pi}$, to be the steady-state vector or the dominant eigenvector that satisfies:

$$\boldsymbol{\pi} A = \boldsymbol{\pi} \tag{4}$$

2.2 Sparse PageRank Definition

Having constructed matrix A we might naïvely attempt to find the PageRank vector of Equation (4) by directly using a power method approach:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} A \tag{5}$$

where $\boldsymbol{x}^{(k)}$ is the k^{th} iterate towards the PageRank vector, $\boldsymbol{\pi}$. However, the web was known to have more than a trillion unique pages in 2008, with several billion new pages being added to this total every day, so it is clear that this is not a practical approach for realistic web graphs [3]. The reason for this is that A is a completely dense matrix, on account of the completely dense teleportation matrix E.

Given the teleportation-matrix density concern, a sparse reduction of the standard equation Equation (4) is typically employed in calculations [18]. The reduction is as follows:

$$\begin{aligned} \boldsymbol{\pi} &= \boldsymbol{\pi} A \\ &= \boldsymbol{\pi}(cP' + (1-c)E) \\ &= c\boldsymbol{\pi} P' + (1-c)\boldsymbol{\pi} E \\ &= c\boldsymbol{\pi} P' + (1-c)\sum_i \boldsymbol{\pi}_i \boldsymbol{p} \\ &= c\boldsymbol{\pi} P' + (1-c)\boldsymbol{p} \end{aligned} \tag{6}$$

where P' is more sparse than the original matrix A.

3 Irreducibility Is Not Required

As given above, it is regularly written in the literature that irreducibility is required for Equation (4) to be well-defined, with a unique steady-state vector. To ensure irreducibility, it is written that a completely dense personalisation vector is required.

Kamvar *et al.* [16] write, "If [the Google matrix] is aperiodic and irreducible, then the Ergodic Theorem guarantees that the stationary distribution of the random walk is unique. In the context of computing PageRank, the standard way of ensuring that [the Google matrix] is irreducible is to add a new set of complete outgoing transitions, with small transition probabilities, to all nodes, creating a complete (and thus an aperiodic and strongly connected) transition graph."

Langville *et al.* [6] write, "[To guarantee] the existence and uniqueness of the PageRank vector, Brin and Page added another rank-one update, this time an irreducibility adjustment in the form of a dense perturbation matrix [E] that creates direct connections between each page."

In the context of an unbiased representation of how we surf the web (without a back button), a completely dense personalisation vector makes sense. But this density requirement poses a difficulty when we start personalising PageRank. If we were to attempt to categorise users—for example, as avid consumers of sports news—or we were to attempt to model how a particular person, or group of people, surfs the web (as per the intuitive justification for PageRank), then it seems clear that we should allow for zero entries in the personalisation vector, zero entries which correspond to those pages to which the particular person will not teleport.

One might argue that this is not the case, that for any person there is a chance, albeit very small, that this particular person teleports to any web page. But it is difficult to justify intuitively why any personalised categorisation of users should necessarily have non-zero probability of teleporting to every web page. Equally, it is difficult to understand why there cannot be a personalised model of a person for which there is at least one zero personalisation vector entry. The argument for a completely dense personalisation vector seems to be based more on a need for theoretical well-definedness than on any force of intuition.

In this subsection, we recall a theorem from [20] from which it follows that complete density is not required for PageRank to be well-defined.

Let us remove the requirement that $\boldsymbol{p}$ be completely dense, and let us have instead that $\boldsymbol{p}$ is just a probability vector. Let R define the set of indices of non-zero entries, $\{p_t > 0 : 1 \leq t \leq n\}$. Now, if we consider Equation (6), then it is clear that independent of the structure of P, every page is connected to each and every page in set R. Indeed, by definition, these are the pages to which the surfer could teleport whilst on any other page. Accordingly, this set R forms part of a single irreducible component of recurrent pages.

Let us now consider those pages which are not part of R. Any such page is either transient, in which case the page has an outgoing path to some page in R but there is no reciprocal path back from pages in R, or such a page has both an outgoing path to a page in R and has a reciprocal path back from a page in R. In the latter case, the page forms part of an irreducible component of pages of which R is a subset.

Let us now recall the following from [20]:

Proposition 1. *Suppose that a DTMC with one-step probability transition matrix T has just one strongly connected subcomponent of recurrent states. This is equivalent to supposing that there is some state which is reachable from all other states—in matrix form, $\exists j \forall i \exists p (T^p_{ij} > 0)$. Then,*

1. *The transient states all have steady-state entries equal to zero.*
2. *The restriction of matrix T to the recurrent states (removing all rows and columns corresponding to transient states) is an irreducible probability transition matrix.*
3. *There is a corresponding unique steady-state distribution.*
4. *The recurrent states all have positive steady-state entries.*

From this proposition, it follows that we require only that $\boldsymbol{p}$ be a probability vector for PageRank to be well-defined.

4 Computing PageRank without Dangling Pages

4.1 Motivating PageRank Computation without Dangling Pages

Eiron *et al.* write that the number of dangling pages is higher than the number of non-dangling pages [4]. Langville and Meyer write that the number of dangling pages relative to non-dangling is growing, and that some sets of crawled pages show percentages of dangling pages reaching 80% [6].

The reason for this high percentage of dangling pages is two-fold: an increasing number of *pseudo*-dangling pages and an increasing number of *real* dangling pages.

Pseudo-dangling pages are pages which are treated as dangling pages because their outlinks have not (yet) been crawled. There are several reasons for this growing number of pages with uncrawled outlinks.

Firstly, over recent years there has been an ever increasing amount of dynamic content on the web, and also links to such content, and the rapid increase has left crawlers incapable of keeping up. Unlike static pages, which are hand-edited HTML, dynamic pages are database-driven. These dynamic pages are limited in number only by what is available in the database, and, potentially, not even then. For example, the number of pages given by a web calendar might be expected to be (nearly) infinite. Even in more mundane examples the size of the database may not provide an upper bound on the number of potential dynamic pages. For example, session IDs, timestamps, and so on, may further expand the potential number of dynamic pages, as ostensibly the same page is treated differently, because it has a different URL or a different embedded timestamp.

Two further reasons for the growing number of uncrawled links are robots.txt/ nofollow and JavaScript. Robots.txt and nofollow are conventions whereby website owners can demarcate certain parts of their sites as not to be crawled. The outlinked pages in a prohibited part of a site are typically still indexed according to anchor text, but as they are not actually crawled, they are treated as dangling pages [4].

JavaScript links are becoming more prevalent, particularly as part of dynamic AJAX websites. Such links are not evaluated by current search engines [21].

Real dangling pages are those from which there really are no outgoing links. These may be HTML pages without links. However, the main reason for the exploding number of *real* dangling pages is the recent push by the research community to move more and more material online: PDF, postscript and PPT files of papers, presentations, theses, and so on.

4.2 Dangling-Page Matrix Yields Scaling

In the literature, there are two linear algebraic treatments of PageRank whereby the dangling page matrix is removed from the PageRank definition. In [6], a somewhat *ad hoc* proof is presented to show that the dangling-page matrix serves only to scale the solution vector. The proof starts with the identity Equation (19). It then proceeds to show that by reformulating this identity we can get the linear algebraic form of Equation (15). It is *ad hoc* because it proceeds by assuming the given identity is the correct one. It gives no clues as to how one might discover this identity in the first instance. In the later [7], the same fact is proved by way of the Sherman-Morrison formula [22].

In this subsection we look to provide a more accessible explanation for the removal of the dangling-page matrix. Our proposal presents PageRank as a special case of a broader class of problem. The kernel of the explanation is the following theorem which allows us to map homogeneous singular linear systems of index 1 to inhomogeneous non-singular linear systems with the same solution vector.

Theorem 2. *Let us define matrix, $V^{(K)} \in C^{n\times n}$ ($K \subseteq \{1,2,\ldots,n\}$):*

$$V^{(K)}_{ij} = \begin{cases} v_i & \text{if } j \in K \\ 0 & \text{if } j \notin K \end{cases}, \tag{7}$$

where $\boldsymbol{v} \in C^n$.

Now suppose that 1 is not an eigenvalue of $(M - V^{(K)})$, for some $M \in C^{n\times n}$, $\boldsymbol{v}$ and K.[1]

Then, we have the following: If 1 is an eigenvalue of M, then it is a simple eigenvalue of M and there is a corresponding right eigenvector $\boldsymbol{x}$ of M which is the unique fixed-point of

$$\boldsymbol{x} = (M - V^{(K)})\boldsymbol{x} + \boldsymbol{v}. \tag{8}$$

Proof: Suppose $\boldsymbol{y} = M\boldsymbol{y}$. Then,

$$\boldsymbol{y} = (M - V^{(K)})\boldsymbol{y} + (\sum_{k\in K} y_k)\boldsymbol{v}. \tag{9}$$

The last equality holds because

$$V^{(K)}\boldsymbol{y} = \begin{pmatrix} v_1 \sum_{k\in K} y_k \\ \vdots \\ v_n \sum_{k\in K} y_k \end{pmatrix} = (\sum_{k\in K} y_k)\boldsymbol{v}. \tag{10}$$

Now, by supposition, 1 is not an eigenvalue of $(M - V^{(K)})$, so:

$$\sum_{k\in K} y_k \neq 0. \tag{11}$$

Also, for all $\alpha \neq 0$, we have the following:

$$\boldsymbol{z} := (M - V^{(K)})\boldsymbol{z} + \alpha\boldsymbol{v} = \alpha(I - (M - V^{(K)}))^{-1}\boldsymbol{v} \tag{12}$$

This is true because, as 1 is not an eigenvalue of $(M - V^{(K)})$:

$$(I - (M - V^{(K)})) \text{ is invertible} \tag{13}$$

[1] Given an irreducible complex matrix, M, with unit spectral radius, an example of suitably chosen $V^{(K)}$ has $\boldsymbol{v}$ equalling the first column of M and $K = \{1\}$.

Accordingly, the fixed-point $\boldsymbol{x}$ of

$$\boldsymbol{x} = (M - V^{(K)})\boldsymbol{x} + \boldsymbol{v} \tag{14}$$

is a scalar multiple of $\boldsymbol{y}$, and thus $\boldsymbol{x}$ is a right eigenvector of M which corresponds to eigenvalue 1. Further, given non-singularity, $\boldsymbol{x}$ is the unique fixed-point, and so 1 is a simple eigenvalue of M.

Let us now recall the standard PageRank definition:

$$\boldsymbol{\pi} = \boldsymbol{\pi}(c(P + D) + (1 - c)E) \tag{15}$$

We know from the Perron–Frobenius theorem together with continuity of spectral radius with respect to matrix entries that if we remove cD from the PageRank matrix to yield $(c(P) + (1 - c)E)$, then the resultant matrix does not have 1 as an eigenvalue. So, by Theorem 2:

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{x}(c(P + D) + (1 - c)E - cD) + \boldsymbol{p} \\ &= \boldsymbol{x}(c(P) + (1 - c)E) + \boldsymbol{p} \\ &= c\boldsymbol{x}P + (1 - c)\boldsymbol{x}E + \boldsymbol{p}, \end{aligned} \tag{16}$$

where $\frac{\boldsymbol{x}}{\sum_i x_i} = \boldsymbol{\pi}$.

Now, we know that

$$(1 - c)\boldsymbol{x}E = (1 - c)\sum_i x_i \boldsymbol{p} \tag{17}$$

where $\sum_i x_i > 0$.

So, from Equation (17), we have that

$$\boldsymbol{x} = c\boldsymbol{x}P + \alpha\boldsymbol{p} \tag{18}$$

where α is some positive real scalar.

If we rewrite this equation as an inhomogeneous non-singular linear system, it is clear that the α coefficient serves only to scale. This allows us to solve instead:

$$\boldsymbol{y} = c\boldsymbol{y}P + \boldsymbol{p} \tag{19}$$

where non-negative vector $\boldsymbol{y}$ is such that $\frac{\boldsymbol{y}}{\sum_i y_i} = \boldsymbol{\pi}$.

So, we have defined a scalar multiple of the PageRank vector $\boldsymbol{\pi}$ which makes no appeal to a dangling-page matrix.

Given the removal of the dangling-page matrix, we find in [6] an iterative procedure for removing dangling pages before we employ power iterations to solve for PageRank.

4.3 Different Dangling and Personalisation Vectors

In [9,10], considerations are presented into generalisations of PageRank which allow the dangling-page vector to differ from the personalisation vector—to account, in particular, for TrustRank [11]. The generalised PageRank definition differs from the standard PageRank in that $D = \boldsymbol{d}\boldsymbol{g}^T$, where dangling-page vector, $\boldsymbol{g}$, is any probability vector. In these two papers, lumpability theory is used to show that, even in this generalised form of PageRank, the dangling-page matrix can be removed.

In this subsection we use Theorem 2 to provide an alternative, linear algebraic reformulation of this generalisation, to remove the dangling-page matrix. We also introduce another theorem, which allows another means of reformulating.

By Theorem 2, reasoning as before, though with the generalised equation,

$$\begin{aligned} \boldsymbol{x} &= \boldsymbol{x}(c(P+D) + (1-c)E - cD) + \boldsymbol{w} \\ &= \boldsymbol{x}(c(P) + (1-c)E) + \boldsymbol{w} \\ &= c\boldsymbol{x}P + (1-c)\boldsymbol{x}E + \boldsymbol{w}, \\ &= c\boldsymbol{x}P + \alpha\boldsymbol{p} + \boldsymbol{w}, \end{aligned} \tag{20}$$

where α is a non-zero scalar.

Now, if we solve the two inhomogeneous nonsingular equations:

$$\boldsymbol{y} = c\boldsymbol{y}P + \boldsymbol{p}; \tag{21}$$

$$\boldsymbol{z} = c\boldsymbol{z}P + \boldsymbol{w}. \tag{22}$$

Then, α is easy to determine by substituting $\boldsymbol{x} = \alpha\boldsymbol{y} + \boldsymbol{z}$ into Equation 20.

In the above, we have shown that the dangling page matrix is not required when solving for generalised PageRank, and we did so by appealing to Theorem 2. An alternative approach would be to appeal to the following theorem. It allows us to split an inhomogeneous nonsingular linear system into two constituent systems. The proof is a simplification of the earlier proof.

Theorem 3. *Let matrix $V^{(K)}$ be defined as before in terms of a vector, $\boldsymbol{v}$, and a set of indices, K. Let $M\boldsymbol{x} = \boldsymbol{w}$ where M is non-singular and $\boldsymbol{w}$ is a vector. Then, if $(M - V^{(K)})$ is non-singular, then*

$$(M - V^{(K)})\boldsymbol{x} = \alpha\boldsymbol{v} + \boldsymbol{w}, \tag{23}$$

for scalar $\alpha = \sum_{i \in K} x_i$.

5 Conclusion and Future Work

In this paper we have revisited the assumption that the personalisation vector needs to be completely dense. We used Proposition 1 to show that this is not the case. In so doing we have presented a generalisation of PageRank which better accords with the intuitive justification given in the literature.

We then introduced Theorem 2. This theorem is broad-ranging in terms of its potential applicability. The theorem may be used to apply novel solution methods to eigenvector problems. For example, given some irreducible complex matrix, M, with unit spectral radius of its modulus equivalent, if we choose $\boldsymbol{v}$ to be the first column of M and if we choose $K = \{1\}$, then the theorem allows us to apply asynchronous solution [15] to solve for the dominant eigenvector of M. Equally, the theorem may be used to improve sparsity patterns or conditioning when using traditional solution methods to such problems. We intend to explore both applications in future work. In this paper, however, the application of—and, indeed, inspiration for—Theorem 2 was the PageRank equation. Applied to this special case, we used the theorem to remove the dangling-page matrix from the PageRank definition.

Finally we considered an extension of the PageRank definition which allows the dangling-page vector to be different from the personalisation vector. We showed how the same linear algebraic framework enables the dangling-page matrix to be removed here. We also suggested an alternative approach via Theorem 3.

References

1. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: Bringing order to the web, Tech. rep. In: Stanford Digital Library Technologies Project (1998)
2. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: Seventh International World-Wide Web Conference, WWW 1998 (1998)
3. Official Google blog, http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html
4. Eiron, N., McCurley, K.S., Tomlin, J.A.: Ranking the web frontier. In: WWW 2004: Proceedings of the 13th international conference on World Wide Web, pp. 309–318. ACM, New York (2004)
5. Avrachenkov, K., Litvak, N.: Decomposition of the Google PageRank and Optimal Linking Strategy, Tech. Rep. RR-5101, INRIA (01 2004)
6. Langville, A.N., Meyer, C.D.: A reordering for the PageRank problem. SIAM J. Sci. Comput. 27, 2112–2120 (2004)
7. Del Corso, G.M., Gullí, A., Romani, F.: Fast PageRank computation via a sparse linear system. Internet Mathematics 2(3)
8. Lee, C.P.-C., Golub, G.H., Zenios, S.A.: A fast two-stage algorithm for computing PageRank and its extensions, Technical report, Stanford InfoLab (2003)
9. Ipsen, I.C.F., Selee, T.M.: PageRank computation, with special attention to dangling nodes. SIAM J. Matrix Anal. Appl. 29(4), 1281–1296 (2007)
10. Lin, Y., Shi, X., Wei, Y.: On computing PageRank via lumping the Google matrix. J. Comput. Appl. Math. 224(2), 702–708 (2009)
11. Gyöngyi, Z., Garcia-Molina, H., Pedersen, J.: Combating web spam with trustrank. In: VLDB 2004: Proceedings of the Thirtieth international conference on Very large data bases, pp. 576–587. VLDB Endowment (2004)
12. Kollias, G., Gallopoulos, E., Szyld, D.B.: Asynchronous iterative computations with web information retrieval structures: The pagerank case, CoRR abs/cs/0606047
13. Cevahir, A., Aykanat, C., Turk, A., Cambazoglu, B.B.: A web-site-based partitioning technique for reducing preprocessing overhead of parallel pagerank computation. In: Kågström, B., Elmroth, E., Dongarra, J., Waśniewski, J. (eds.) PARA 2006. LNCS, vol. 4699, pp. 908–918. Springer, Heidelberg (2007)

14. Bradley, J.T., de Jager, D., Knottenbelt, W.J., Trifunovic, A.: Hypergraph Partitioning for Faster Parallel PageRank Computation. In: Bravetti, M., Kloul, L., Zavattaro, G. (eds.) EPEW/WS-EM 2005. LNCS, vol. 3670, pp. 155–171. Springer, Heidelberg (2005)
15. Chazan, D., Miranker, W.L.: Chaotic relaxation. Linear Algebra and Its Applications 2, 199–222 (1969)
16. Kamvar, S.D., Haveliwala, T.H., Manning, C.D., Golub, G.H.: Extrapolation methods for accelerating PageRank computations. In: Proceedings of the 12th Int. World Wide Web Conference (2003)
17. Berkhin, P.: A survey on PageRank computing. Internet Mathematics 2, 73–120 (2005)
18. Langville, A.N., Meyer, C.D.: Deeper inside PageRank. Internet Mathematics 1(3), 335–380 (2004)
19. Haveliwala, T., Kamvar, S.: The second eigenvalue of the Google matrix, Technical Report 2003–20, Stanford InfoLab (2003)
20. Berman, A., Plemmons, R.J.: Nonnegative matrices in the mathematical sciences. Academic Press, New York (1979)
21. Thurow, S., Sullivan, D.: Search Engine Visibility. Pearson Education, London (2002)
22. Golub, G.H., van Loan, C.F.: Matrix Computations, 3rd edn. Johns Hopkins University Press, Baltimore (1996)

Training Data Cleaning for Text Classification

Andrea Esuli and Fabrizio Sebastiani

Istituto di Scienza e Tecnologia dell'Informazione
Consiglio Nazionale delle Ricerche
Via Giuseppe Moruzzi 1 – 56124 Pisa, Italy
{andrea.esuli,fabrizio.sebastiani}@isti.cnr.it

Abstract. In text classification (TC) and other tasks involving supervised learning, labelled data may be scarce or expensive to obtain; strategies are thus needed for maximizing the effectiveness of the resulting classifiers while minimizing the required amount of training effort. *Training data cleaning* (TDC) consists in devising ranking functions that sort the original training examples in terms of how likely it is that the human annotator has misclassified them, thereby providing a convenient means for the human annotator to revise the training set so as to improve its quality. Working in the context of boosting-based learning methods we present three different techniques for performing TDC and, on two widely used TC benchmarks, evaluate them by their capability of spotting misclassified texts purposefully inserted in the training set.

1 Introduction

In many applicative contexts involving supervised learning, labelled data may be scarce or expensive to obtain. In such situations, once we have trained the classifiers with the available training data (and tested them on the test data, and/or applied them to the unlabelled data that need to be classified), we are often left with the issue of how to improve the effectiveness of the existing classifiers, given that the amount of humanpower needed to perform further labelling is limited. One potential solution is to apply (computer-assisted) *training data cleaning* (TDC). TDC techniques attempt to minimize the additional effort required from human annotators. Indeed, training data often contain misclassified items, sometimes as a result of lack of experience on the part of the junior annotators who have performed the labelling, sometimes as a result of tight time constraints under which the labelling activity has been performed. A good TDC technique top-ranks the training examples with the highest likelihood of being misclassified, which allows the human annotator to improve the quality of the training set by double-checking the labels attached to the training examples, starting with the ones most likely to be erroneous, and working down the ranked list until s/he sees fit. We present three different techniques for performing TDC in TC, and test them using a boosting-based supervised learning device that generates confidence-rated predictions. The reason we are using this device is that it has two features that allow us to exemplify our TDC techniques particularly

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 29–41, 2009.

well, i.e., (i) it allows for a notion of confidence in the classifier's classification decisions; and (ii) the classifier it generates is actually a classifier committee.

2 Preliminaries

This work attempts to identify good TDC techniques for *text classification* (aka *text categorization* – TC), and for *multi-label* text classification (MLTC) in particular. Given a set of textual documents D and a predefined set of *classes* (aka *labels*, or *categories*) $C = \{c_1, \ldots, c_m\}$, MLTC can be defined as the task of estimating an unknown *target function* $\Phi : D \times C \rightarrow \{-1, +1\}$, that describes how documents ought to be classified, by means of a function $\hat{\Phi} : D \times C \rightarrow \{-1, +1\}$ called the *classifier*[1]; here, $+1$ and -1 represent membership and non-membership of the document in the class. As usual, we accomplish MLTC by generating m independent binary classifiers $\hat{\Phi}^j : D \rightarrow \{-1, +1\}$, one for each $c_j \in C$, entrusted with the task of deciding whether a document belongs or not to class c_j.

As the learning device we use a boosting-based learner, called MP-Boost [1]; MP-Boost is a variant of AdaBoost.MH [2] optimized for multi-label settings, which has been shown in [1] to obtain considerable effectiveness improvements with respect to AdaBoost.MH.

MP-Boost works by iteratively generating, for each class c_j, a sequence $\hat{\Phi}^j_1, \ldots, \hat{\Phi}^j_S$ of classifiers (called *weak hypotheses*). A weak hypothesis is a function $\hat{\Phi}^j_s : D \rightarrow \mathbf{R}$, where D is the set of documents and $\mathbf{R}$ is the set of real numbers. The sign of $\hat{\Phi}^j_s(d_i)$ (denoted by $sgn(\hat{\Phi}^j_s(d_i))$) represents the binary decision of $\hat{\Phi}^j_s$ on whether d_i belongs to c_j, i.e. $sgn(\hat{\Phi}^j_s(d_i)) = +1$ (resp., -1) means that d_i is believed to belong (resp., not to belong) to c_j. The absolute value of $\hat{\Phi}^j_s(d_i)$ (denoted by $|\hat{\Phi}^j_s(d_i)|$) represents instead the confidence that $\hat{\Phi}^j_s$ has in this decision, with higher values indicating higher confidence.

At each iteration s MP-Boost tests the effectiveness of the most recently generated weak hypothesis $\hat{\Phi}^j_s$ on the training set, and uses the results to update a distribution D^j_s of weights on the training examples. The initial distribution D^j_1 is uniform. At each iteration s all the weights $D^j_s(d_i)$ are updated, yielding $D^j_{s+1}(d_i)$, so that the weight assigned to an example correctly (resp., incorrectly) classified by $\hat{\Phi}^j_s$ is decreased (resp., increased). The weight $D^j_{s+1}(d_i)$ is thus meant to capture how ineffective $\hat{\Phi}^j_1, \ldots, \hat{\Phi}^j_s$ have been in guessing the correct *c_j-assignment* of d_i (denoted by $\Phi^j(d_i)$), i.e., in guessing whether training document d_i belongs to class c_j or not. By using this distribution, MP-Boost generates a new weak hypothesis $\hat{\Phi}^j_{s+1}$ that concentrates on the examples with the highest weights, i.e. those that had proven harder to classify for the previous weak hypotheses. The overall prediction on whether d_i belongs to c_j is obtained as a sum $\hat{\Phi}^j(d_i) = \sum_{s=1}^{S} \hat{\Phi}^j_s(d_i)$ of the predictions of the weak hypotheses. The final classifier $\hat{\Phi}^j$ is thus a *committee* of S classifiers, each classifier casting a

[1] Consistently with most mathematical literature we use the caret symbol (ˆ) to indicate estimation.

weighted vote (the vote being the binary decision $sgn(\hat{\Phi}^j_s(d_i))$, the weight being the confidence $|\hat{\Phi}^j_s(d_i)|$) on whether d_i belongs to c_j. For the final classifier $\hat{\Phi}^j$ too, $sgn(\hat{\Phi}^j(d_i))$ represents the binary decision as to whether d_i belongs to c_j, while $|\hat{\Phi}^j(d_i)|$ represents the confidence in this decision.

3 Three Techniques for Training Data Cleaning

In the following, by a *TDC technique* we will mean a technique that, given a training set Tr and a class c_j, produces a ranking $r_j(Tr)$ in which the elements of Tr are sorted in decreasing order of their likelihood of being mislabelled for c_j. Different techniques correspond to different ways of estimating this likelihood.

We now present three alternative TDC techniques. For each $c_j \in C$, the first technique (that we dub *the confidence-based technique* – CON, in short) consists in (i) training the classifier $\hat{\Phi}^j$ on Tr; (ii) reclassifying Tr by means of $\hat{\Phi}^j$; and (iii) ranking Tr in increasing order of $\hat{\Phi}^j(d_i) \cdot \Phi^j(d_i)$ value. Note that, while $\Phi^j(d_i)$ is a value in {-1,+1}, $\hat{\Phi}^j(d_i)$ is a value in $(-\infty, +\infty)$, so $\hat{\Phi}^j(d_i) \cdot \Phi^j(d_i)$ is also in $(-\infty, +\infty)$. A positive (resp., negative) value of $\hat{\Phi}^j(d_i) \cdot \Phi^j(d_i)$ indicates correct (resp., incorrect) classification, while a high (resp., low) absolute value of $\hat{\Phi}^j(d_i) \cdot \Phi^j(d_i)$ indicates that this classification decision has been taken with high (resp., low) confidence. CON thus corresponds to (a) top-ranking the examples $d_i \in Tr$ that $\hat{\Phi}^j$ has misclassified, in decreasing order of the confidence $|\hat{\Phi}^j(d_i)|$ with which $\hat{\Phi}^j$ has taken its decision, and (b) appending to this list the examples $d_i \in Tr$ that $\hat{\Phi}^j$ has correctly classified, in increasing order of the confidence $|\hat{\Phi}^j(d_i)|$. The rationale of this technique is that, if $\hat{\Phi}^j$ has misclassified a training example d_i with high confidence, this means that the c_j-assignment made to d_i by the human annotator is highly at odds with the c_j-assignments that the human annotator has made for the other training examples. This indicates that the human annotator may well have misclassified d_i for c_j.

For each $c_j \in C$, the second technique (that we dub *the nearest neighbours technique* – NN) consists in ranking the training examples in terms of how inconsistent their c_j-assignment is with the c_j-assignments of their k nearest neighbours, for a predefined k. More formally, this consists in (i) computing, for each $d_i \in Tr$, the value

$$\zeta(d_i, c_j) = \sum_{d_z \in Tr_k(d_i)} sim(d_i, d_z) \cdot \Phi^j(d_z) \tag{1}$$

where $sim(\cdot, \cdot)$ denotes a measure of similarity between documents and $Tr_k(d_i)$ denotes the k training examples most similar to d_i; and (ii) ranking Tr in increasing order of $\zeta(d_i, c_j) \cdot \Phi^j(d_i)$ value. For class c_j, the examples d_i with c_j-assignments highly consistent with the c_j-assignments of their neighbours will have high $\zeta(d_i, c_j) \cdot \Phi^j(d_i)$ values, which means that the top-ranked examples (which are the ones with the lowest $\zeta(d_i, c_j) \cdot \Phi^j(d_i)$ values) will be the ones with c_j-assignments most dissimilar from those of their closest neighbours.

Equation (1), of course, is that of the standard distance-weighted k-NN learning device, the only difference being that, while in the standard case $\Phi^j(d_z)$ ranges on {0,1}, in our case it ranges on {-1,+1}, which means that neighbours with a negative c_j-assignment weigh negatively on $\zeta(d_i, c_j)$.

For each $c_j \in C$, the third technique (that we dub *the committee-based technique* – COM) consists in (i) training the classifier $\hat{\Phi}^j$ on Tr; (ii) reclassifying Tr by means of $\hat{\Phi}^j$; and (iii) ranking Tr in increasing order of $\Delta(\hat{\Phi}^j(d_i)) \cdot sgn(\hat{\Phi}^j(d_i)) \cdot \Phi^j(d_i)$ value, where $\Delta(\hat{\Phi}^j(d_i))$ is a measure of the *disagreement* among the S members of $\hat{\Phi}^j$ on whether d_i belongs to c_j or not. This technique is based on the intuition that the examples most in need of double-checking are the ones which $\hat{\Phi}^j$ has misclassified (i.e., are such that $sgn(\hat{\Phi}^j(d_i)) \cdot \Phi^j(d_i) = -1$) *with the most widespread agreement* among its S members. In other words, if the information that a training example provides to the training process is so inconsistent with that provided by the other training data, as to have the members of the generated classifier committee misclassify the example, and with widespread agreement, then it is likely that the example might be mislabelled. This technique will thus top-rank the training examples that the committee has misclassified and on which the S members of the committee agree most. The key difference between the first technique (CON) and this technique is that here the confidence that a classifier committee has in a certain decision is taken to coincide with the level of (weighted) agreement among its members, and not with the (weighted) sum of the individual opinions. As a measure of disagreement among the S members of the committee we have chosen to use *standard deviation* (denoted σ). This is a natural choice, given that the values $\hat{\Phi}^j_1(d_i), \ldots, \hat{\Phi}^j_S(d_i)$ are real numbers: standard deviation thus allows to measure disagreement by taking into account not only the polarity $sgn(\hat{\Phi}^j_s(d_i))$ of each member's decision, but also its confidence level $|\hat{\Phi}^j_s(d_i)|$, so that two members with views of different polarity are taken to disagree more if they are highly confident in their views, and less if they are not.

Actually, there is a fourth technique (that we dub *the distribution-based technique* – DIS) that might come to mind. For each $c_j \in C$, this technique consists in (i) training the classifiers $\hat{\Phi}^j$ on Tr, and (ii) ranking the examples $d_i \in Tr$ in decreasing order of the $D^j_S(d_i)$ value that MP-Boost has produced as a side effect of the learning process. The rationale of this technique is that, since the training examples that maximize $D^j_S(d_i)$ are the ones that have turned out the most difficult to make sense of during the boosting iterations, they are thus the ones whose c_j-assignment is most highly at odds with the c_j-assignment of the other training examples. The problem with the DIS technique is that it turns out to be equivalent to our first technique (CON), in the sense that CON and DIS always generate identical rankings, a fact that had never been noted in the literature[2]. The only advantage that DIS provides over CON is thus that there is

[2] We discovered this fact experimentally in the course of this work. A conversation with Robert Schapire, one of the "fathers" of boosting, later revealed that, while this phenomenon had never been observed before, an *a posteriori* justification can be found for it in the theory that underlies the AdaBoost.MH algorithm, of which MP-Boost is a variant.

no need to reclassify the training examples by means of $\hat{\Phi}^j$, since the information needed for ranking is already available after training has occurred.

Before discussing the experiments it is worthwhile noting that, although we have described these techniques in the context provided by a boosting-based learner which generates confidence-rated predictions, all of these techniques can be used also in connection with other learning devices. More specifically, CON only needs the classifier to return a score of confidence in its own decision, NN has no specific requirements, and COM requires the classifier to consist of a committee of classifiers. Moreover, the discussed equivalence between CON and DIS has the practical consequence of making available a technique equivalent to DIS to learning devices not based on boosting.

4 Experiments

In order to test our TDC techniques we use a standard MLTC dataset $\Omega = \langle Tr, Te \rangle$ split into a training set Tr and a test set Te. We assume that Tr contains no misclassified examples, and we simulate the presence of misclassified training examples by artificially "perturbing" a small number m of training examples; we call the value $p = \frac{m}{|Tr|}$ the *perturbation ratio*. In what follows, "perturbing a training example d_i for class c_j" means changing its c_j-assignment, from positive to negative (in this case we call d_i a *false negative for* c_j) or from negative to positive (a *false positive*); by $\widehat{Tr}$ we denote the training set after perturbation.

We test two different perturbation techniques, which we call *random perturbation* (RP) and *targeted perturbation* (TP). As the name implies, in RP the training examples to perturb are picked at random from Tr. The same training examples (x% of the entire lot) are perturbed for all classes $c_j \in C$. TP is instead obtained by (i) training the classifiers $\hat{\Phi}^j$ on Tr, (ii) reclassifying Tr by means of them, (iii) ranking, for each $c_j \in C$, the reclassified examples in increasing order of the confidence $|\hat{\Phi}^j(d_i)|$ that $\hat{\Phi}^j$ had in classifying them, and (iv) perturbing the top-ranked ones, in number equal to x% of the training examples. The rationale of this technique is that the training examples that $\hat{\Phi}^j$ classifies with low confidence are more likely to be "borderline" examples for c_j. As a result, these examples are the ones that, should they be manually labelled, would have the highest probability of being misclassified (either due to lack of experience or to lack of adequate time) by a human annotator. In other words, while RP simulates the perturbation of a training set that might derive from, say, file corruption, TP simulates the perturbation that might derive from lack of experience, or lack of care, on the part of the human annotator who has labelled the training set. Unlike in RP, in TP we allow different training examples to be perturbed for different classes $c_j \in C$, since the same document might be controversial, or "borderline", for one class but not for others.

In order to determine which among the three TDC techniques of Section 3 is the best we will measure how good each technique is at ranking $\widehat{Tr}$ in such a way that the perturbed training examples are placed at the top of the ranking. To this end, it seems natural to adopt one of the measures routinely used for evaluating ad-hoc (ranked) retrieval. Of course, ad-hoc retrieval is all about ranking the "good" (i.e., relevant to the information need) examples higher than the bad ones, while TDC aims at ranking the "bad" (i.e., likely misclassified) examples higher than the good ones; but this is of course an inessential difference.

As a measure of ranking quality we will choose *mean average precision* (MAP), which in our context is defined as follows. Let $r_j(\widehat{Tr})$ be the ranking for class c_j, realized according to TDC technique r, of the perturbed training set $\widehat{Tr}$, and let $[r_j(\widehat{Tr})]_k$ be a binary predicate that returns 1 if the example at the k-th position in $r_j(\widehat{Tr})$ is perturbed for class c_j, and 0 otherwise. We define the *precision at n of* $r_j(\widehat{Tr})$ as

$$P_n(r_j(\widehat{Tr})) = \frac{1}{n}\sum_{k=1}^{n}[r_j(\widehat{Tr})]_k \tag{2}$$

We then define the *average precision of* $r_j(\widehat{Tr})$ as

$$AP(r_j(\widehat{Tr})) = \frac{\sum_{k=1}^{|\widehat{Tr}|} P_k(r_j(\widehat{Tr})) \cdot [r_j(\widehat{Tr})]_k}{\sum_{k=1}^{|\widehat{Tr}|}[r_j(\widehat{Tr})]_k} \tag{3}$$

The *mean average precision* (MAP) of technique r on $\widehat{Tr}$ is finally defined as

$$MAP(r(\widehat{Tr})) = \frac{1}{|C|}\sum_{c_j \in C} AP(r_j(\widehat{Tr})) \tag{4}$$

Aside from a measure of TDC effectiveness we also need a measure of MLTC effectiveness, so as to determine the effectiveness gains in classification obtained if TDC is performed. For this purpose we have used the well-known F_1 function, in both its microaveraged (F_1^μ) and macroaveraged (F_1^M) variants.

Section 4.2 reports the results of our experiments with the three TDC techniques of Section 3, the two different perturbation techniques, different perturbation ratios, and different datasets Ω.

4.1 The Datasets

In our experiments we have used the Reuters-21578 and RCV1-v2 datasets. Reuters-21578 is probably still the most widely used benchmark in MLTC research. It consists of a set of 12,902 news stories, partitioned (according to the "ModApté" split we have adopted) into a training set of 9,603 documents and a test set of 3,299 documents. The documents are labelled by 118 categories; in our experiments we have restricted our attention to the 115 categories with at least one positive training example. Reuters Corpus Volume 1 version 2 (RCV1-v2) is a more recent MLTC benchmark made available by Reuters and consisting of 804,414 news stories produced by Reuters from 20 Aug 1996 to 19 Aug 1997. In our experiments we have used the "LYRL2004" split, defined in [3], in which the (chronologically) first 23,149 documents are used for training

Table 1. Mean average precision (MAP) of the three TDC techniques (CON, NN, COM) on the full set of classes (top 4 rows) and on the 30 most infrequent classes (bottom 4 rows) of Reuters-21578 (left) and RCV1-v2 (right). **Boldface** indicates the best performer for a given combination of perturbation ratio (p), perturbation method, and dataset.

		Reuters-21578						RCV1-v2					
		Random			Targeted			Random			Targeted		
	p	CON	NN	COM	CON	NN	COM	CON	NN	COM	CON	NN	COM
Full Set	.001	**.596**	.458	.305	**.510**	.369	.152	.232	**.238**	.072	**.357**	.082	.125
	.010	.653	**.771**	.517	**.608**	.525	.206	**.752**	.542	.566	**.519**	.376	.194
	.050	**.968**	.907	.808	**.677**	.621	.301	**.927**	.777	.801	**.672**	.512	.417
	.100	**.973**	.961	.874	**.665**	.634	.449	**.945**	.865	.804	**.658**	.593	.520
30 Infr	.001	.748	**.790**	.401	.648	**.681**	.100	.222	**.225**	.099	**.323**	.101	.104
	.010	.674	**.966**	.599	.581	**.670**	.153	**.702**	.476	.533	**.435**	.375	.275
	.050	.982	**.992**	.812	.647	**.701**	.268	**.896**	.716	.747	**.608**	.427	.479
	.100	.981	**.985**	.886	**.673**	.651	.455	**.919**	.845	.760	**.613**	.523	.588

and the other 781,265 are used for testing. Of the 103 "Topic" categories, in our experiments we have restricted our attention to the 101 categories with at least one positive training example.

In all the experiments discussed in this paper stop words have been removed, punctuation has been removed, all letters have been converted to lowercase, numbers have been removed, and stemming has been performed by means of Porter's stemmer. Word stems are thus our indexing units; since MP-Boost requires binary input, only their presence/ absence in the document is recorded, and no weighting is performed.

4.2 Results and Discussion

Table 1 reports MAP values obtained by ranking the perturbed training sets by means of the three TDC techniques (CON, NN, COM). Results are reported for the full set of classes and for the 30 most infrequent classes of both Reuters-21578 and RCV1-v2. The reason we pay special attention to the most *infrequent* classes is that they are usually the classes for which standard supervised learning techniques produce the lowest classification accuracy, which means that they are the classes which are most in need of effectiveness improvement, by TDC or other technique: a user might typically engage in TDC for these highly problematic classes and forget about the classes for which high enough accuracy has already been achieved.

In all the experiments MP-Boost has been run with a number S of iterations fixed to 1,000. For the NN technique, as the $sim(\cdot,\cdot)$ measure of inter-document similarity we have used the cosine of the angle between the $tfidf$ vectors of the two documents. For the same technique we have used the value $k = 45$, since in using k-NN as a learning device for TC Yang [4] has found this value to yield the best effectiveness and has found negligible differences between values of $k \in [30, 65]$; we defer careful optimization of the k parameter to further work.

A "trivial" baseline to the results of Table 1 is the expected MAP value of the random ranker (RR). Detailed combinatorial analysis shows that this is equal to

$$MAP(RR(\Omega)) = \frac{m-1}{n-1} + \frac{(n-m)H_n}{n(n-1)} \tag{5}$$

where m is the number of relevant (in our case: misclassified) examples in the document set Ω, n is the total number of examples in Ω, and H_n denotes the n-th harmonic number (i.e., $H_n = \sum_{k=1}^{n} \frac{1}{k}$). Actual computation of this formula shows that $MAP(RR(\Omega))$ is approximated by $\frac{m}{n}$ (and in an especially accurate way for large values of n), which in our case coincides with the perturbation ratio $p = \frac{m}{|Tr|}$. Since for all of our datasets and perturbation ratios approximating Equation (5) to the third decimal digit exactly yields p, the first column of Table 1 also indicates the trivial baseline for the experiments in the corresponding row.

There are several insights that can be gained from observing the results of Table 1. The first observation is that, since picking training examples at random is the only method one can adopt when wanting to perform TDC, unless equipped with a specific TDC technique such as CON, NN or COM, the improvements that the three TDC techniques display in Table 1 over the baseline of Column 1 is noteworthy.

A second observation is that, with few exceptions and all other things being equal, each technique performs better for random perturbation than for targeted perturbation. This is intuitive, since misclassified training examples inserted at random in the training set tend to be easier to spot; conversely, in targeted perturbation we corrupt the label of borderline examples, which are then much more difficult to identify for *any* technique.

The third observation is that, among the three competing TDC techniques, while there is no clear winner, there is certainly one clear loser, namely, the COM technique, which in almost all situations obtains results inferior (and often radically so) to CON and NN. We think that the reason for the bad performance of COM may be found in the fact that MP-Boost generates a committee of classifiers that are not independent of each other. Indeed, each member $\hat{\Phi}_s^j$ of the committee strongly depends on the previously generated member $\hat{\Phi}_{s-1}^j$, since the former is generated according to the distribution resulting from applying $\hat{\Phi}_{s-1}^j$ to Tr. As a consequence, agreement is probably not something one could reasonably expect from the members of *this* kind of committee, since sharp disagreement may derive from reasons different from a bad label, such as the different emphasis that the different members place, by construction, on a given training example.

Leaving COM aside, we may observe that neither CON nor NN systematically outperform the other. CON tends to be the better technique on the RCV1-v2 dataset, while the situation is less clearcut on Reuters-21578; similarly, CON tends to outperform NN on the full set of classes of each dataset, while when we analyse the behaviour of the two techniques on the 30 most infrequent classes of each dataset there is no clear winner. All in all, both techniques turn out to be respectable contenders, often achieving (sometimes surprisingly) high MAP values in absolute terms.

Table 2. Micro- and macro-averaged F_1 values for the full set of classes (top 5 rows) and for the 30 most infrequent classes (bottom 5 rows) of Reuters-21578 (left) and RCV1-v2 (right) after random or targeted perturbation

		Reuters-21578				RCV1-v2			
		Random		Targeted		Random		Targeted	
	p	F_1^μ	F_1^M	F_1^μ	F_1^M	F_1^μ	F_1^M	F_1^μ	F_1^M
Full Set	.000	.852	.606	.852	.606	.572	.423	.572	.423
	.001	.822	.356	.821	.448	.557	.368	.558	.354
	.010	.583	.227	.632	.254	.348	.224	.441	.324
	.050	.138	.074	.209	.094	.105	.096	.211	.160
	.100	.064	.047	.116	.061	.050	.064	.137	.107
30 Infr	.000	.373	.245	.373	.245	.164	.062	.164	.062
	.001	.190	.114	.139	.137	.102	.044	.038	.035
	.010	.038	.036	.056	.052	.025	.024	.063	.039
	.050	.004	.004	.011	.011	.006	.005	.015	.014
	.100	.002	.002	.006	.005	.005	.003	.010	.008

A fourth insight we can gain by looking at Table 1 is that MAP tends to increase with the perturbation ratio p, and may reach extremely high values for high values of p. This is very good news, since this means that if we have reasons to believe that our training set is extremely low-quality, we know that our time in cleaning it will not be wasted, since these techniques will place almost all the bad examples near the top of the ranking.

Table 2 reports instead the micro- and macro-averaged F_1 values obtained before and after perturbation; this is an indication of the improvement in classification effectiveness one obtains by performing TDC if the original training set contains noise at the perturbation ratios indicated. Results are reported for the full set of classes and for the 30 most infrequent classes of our two datasets.

One insight that this table allows to gain is that random perturbation is usually more damaging to effectiveness than targeted perturbation, and this fact tends to become evident as the perturbation rate increases. That targeted perturbation may have less disruptive effects is intuitive, since TP introduces mislabellings on documents that are likely borderline examples anyway, i.e., documents that two human annotators might legitimately label in different ways. Mislabelling them may hurt classification accuracy in the thin region of document space close to the surface that separates the positives from the negatives, but does not affect accuracy elsewhere. Conversely, random perturbation may have effects anywhere in document space, and may seriously mislead the classifiers even on cases that would be clearcut otherwise.

A second observation that immediately jumps to the eye is that the decrease in effectiveness deriving from perturbation is noteworthy even for very modest perturbation rates (e.g., .001), and becomes disastrous even for slightly less modest ones (e.g., .010). For instance, for a .001 targeted perturbation rate removing the mislabellings from the Reuters-21578 training set makes F_1^μ jump

- from .821 to .852 for the full set of classes. This is a 3% relative improvement, that in the '90s has taken years of improvement in TC technology to achieve.

This shows that one mislabelled document in a thousand can single-handedly defy the efforts of many TC researchers at improving effectiveness;

- from .139 to .373 for the 30 most infrequent classes, a 168% relative improvement. It is not hard to see why the effect of even a few misclassified training examples on the classification accuracy for infrequent classes can be so large. Given a class with very few positive training examples, mislabelling even one or a handful negatives as positives can severely corrupt the set of positive training examples, while mislabelling even one or a handful of positives as negatives has the double effect of depleting the already slim set of positive examples and confusing the learner by presenting it with negative training documents that are very similar to the remaining positive ones.

These two observations hold to an even higher degree for F_1^M; similar observations also hold for random perturbation and RCV1-v2. For reasons of space we do not separately report the results on the $(|C|-30)$ most frequent classes of our two datasets. In a nutshell, on these classes the decrease in F_1^μ is very similar to the decrease on the full set of classes (since F_1^μ is mostly influenced by the behaviour on the most frequent classes), while the decrease in F_1^M is smaller than the decrease in the full set of classes (since F_1^M is equally influenced by all the classes in C).

Note that Table 2 only gives us a picture of the improvement that might be obtained by cleaning the *entire* training set. Aside from probably being infeasible in many real-world situations, this is something that would defy the purpose of the TDC techniques we have presented. A study should thus be performed that, for any combination of TDC technique, perturbation method, perturbation ratio, and dataset, plots the effectiveness of the classifiers generated after TDC has been performed, as a function of K, the number of top-ranked training examples that the human annotator has double-checked for misclassifications. This is obviously a daunting experimentation, since for each such combination and each value of K the classifiers should be retrained from scratch and the test examples should be reclassified anew. In Table 3 we provide a sample such

Table 3. Micro- and macro-averaged F_1 values for the full set of classes (top 5 rows) and for the 30 most infrequent classes (bottom 5 rows) of Reuters-21578 with classifiers trained after performing TDC by means of the CON technique with $K = 100$

		Random		Targeted	
	p	F_1^μ	F_1^M	F_1^μ	F_1^M
Full Set	.000	.852	.606	.852	.606
	.001	.846	.466	.850	.498
	.010	.749	.399	.780	.412
	.050	.607	.252	.632	.312
	.100	.173	.090	.213	.208
30 Infr	.000	.373	.245	.373	.245
	.001	.260	.187	.202	.197
	.010	.219	.174	.201	.183
	.050	.077	.064	.080	.072
	.100	.013	.013	.020	.019

experiment, in which for different perturbation methods and ratios we test the effectiveness values resulting, on Reuters-21578, from performing TDC by the CON technique and "un-perturbing" the perturbed documents found at the top $K = 100$ positions in the ranking. For instance, with targeted perturbation and $p = .001$, the MAP value of .510 that CON obtains guarantees (see Table 1) that F_1^μ, that perturbation had brought down from .852 to .821 (see Table 2), jumps back to .850, and that F_1^M, that perturbation had brought down from .606 to .448, jumps back to .498. All these results are indicative of the fact that TDC is an important and cost-effective way of improving accuracy for all the datasets of less-than-perfect quality of annotation.

5 Related Work

Several works have used TDC in learning tasks other than TC, especially within the realm of computational linguistics. Some of these works use task-independent TDC techniques while others do not. Among the former, [5,6] use the DIS technique discussed at the end of Section 3, while [7] uses a technique analogous to DIS that exploits the characteristics of SVMs. Other works use instead task-specific techniques; for instance, in a POS-tagging application [8] top-ranks multiple occurrences of the same word that have been classified with different parts of speech in similar linguistic contexts, a technique that is obviously applicable to POS-tagging only and not to tasks such as TC. To the best of our knowledge the only work that deals with TDC in the context of TC is [9]. The proposed method consists in training an SVM, removing from the training set the support vectors that the SVM has identified, training a naive Bayesian classifier on the modified training set, and reclassifying the removed support vectors with this classifier, declaring mislabelled the support vectors whose original label does not match the newly assigned label. The intuition behind this technique is that if a training example has a wrong c_j-assignment, then it likely ends up being a support vector for the generated classifier. Unlike our techniques, this technique is strictly learner-dependent, since it only works with SVMs as learners. Additionally, the method is only limited to cleaning the support vectors; our method examines (and ranks) instead the entire training set; as a result, experimentally comparing the technique of [9] with ours would be problematic.

All of the works above adopt an *a posteriori* evaluation methodology, i.e., they perform no training set perturbation, and evaluate their techniques by ranking the original training sets and then asking human annotators to look for misclassified examples throughout the first k ranks, thus reporting precision-at-k results. We prefer the *a priori* evaluation methodology, since (i) it allows us to work with different perturbation ratios, thus addressing the fact that different applications may be characterized by different levels of quality in their data; (ii) it is exempt from evaluator bias, which the *a posteriori* methodology especially suffers from when (as is frequently the case) it is the authors themselves that engage in post-checking the results; (iii) it allows to compute MAP, while the *a posteriori* methodology only allows to compute precision for a specific, usually low value of k (i.e., the misclassified items from the $(k+1)$-st position onwards have no impact on the evaluation); and (iv) it allows one researcher to replicate the results of the other, while the *a posteriori* methodology does not.

Finally, let us note that the COM technique is somehow reminiscent of the *query-by-committee* active-learning method (see e.g., [10]), in which *unlabelled* examples (and not labelled ones, as in our case) are ranked for human annotation in decreasing order of the disagreement among a committee of classifiers that try to classify them. As a measure of disagreement, [10] uses entropy. We have instead proposed using standard deviation, since entropy can only take into account the binary decisions of the various classifiers, and not the real-valued confidence in their decision; conversely, standard deviation can naturally account for predictions expressed as real numbers, and is thus a better fit in our case.

6 Conclusions

We have tested three techniques for TDC on two popular MLTC benchmarks, checking their ability at spotting and top-ranking a set of training examples whose class assignment we have purposefully corrupted for experimental reasons. This experimental protocol allows to conveniently study *in vitro* the behaviour of these TDC techniques, and to precisely measure the relative merits of the various techniques by means of evaluation measures, such as MAP, standard in the field of ranked retrieval. Studying three TDC techniques with two different perturbation models, at five different perturbation levels, across two datasets (one of which consisting of more than 800,000 documents), and studying both the quality of the resulting rankings *and* the increase in effectiveness that carrying out TDC may bring about, our work probably qualifies as the first truly-large scale experimentation of TDC in either computational linguistics or IR.

Our experimental results show that two techniques, the confidence-based technique and the nearest neighbours technique, achieve good MAP values across different settings deriving from the choice of different datasets, different class frequency, different perturbation ratios, and different types of perturbation, but also show that neither one clearly outperforms the other. A further result of this paper is that a fourth technique, which had been proposed before and which was specific to boosting-based learners, is equivalent to the confidence-based technique proposed here, which is instead applicable to all learners equipped with a notion of confidence in the classification decision. Our results also show that TDC is important, since they show that even a single misclassified example in a thousand training examples can bring about deteriorations in effectiveness that are simply noteworthy in general, and are no less than dramatic for the most infrequent classes and for macroaveraged F_1 in general.

Note also that TDC techniques are important not only for *training data* cleaning, but also for cleaning generic sets of labelled text: the very same techniques discussed here might be applied by a human annotator in order to clean a manually annotated text corpus (e.g., the entire RCV1-v2), regardless of the fact that the entire corpus is then going to be used for training a text classifier. For instance, this is useful for cleaning *test* sets, since incorrectly labelled test examples prevent the accurate measurement of effectiveness, but it is also useful for cleaning labelled datasets produced within organizations that entirely rely on manual classification.

References

1. Esuli, A., Fagni, T., Sebastiani, F.: MP-Boost: A multiple-pivot boosting algorithm and its application to text categorization. In: Crestani, F., Ferragina, P., Sanderson, M. (eds.) SPIRE 2006. LNCS, vol. 4209, pp. 1–12. Springer, Heidelberg (2006)
2. Schapire, R.E., Singer, Y.: Boostexter: A boosting-based system for text categorization. Machine Learning 39(2/3), 135–168 (2000)
3. Lewis, D.D., Yang, Y., Rose, T.G., Li, F.: RCV1: A new benchmark collection for text categorization research. Journal of Machine Learning Research 5, 361–397 (2004)
4. Yang, Y.: An evaluation of statistical approaches to text categorization. Information Retrieval 1(1/2), 69–90 (1999)
5. Abney, S., Schapire, R.E., Singer, Y.: Boosting applied to tagging and PP attachment. In: Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC 1999), College Park, US, pp. 38–45 (1999)
6. Shinnou, H.: Detection of errors in training data by using a decision list and Adaboost. In: Proceedings of the IJCAI 2001 Workshop on Text Learning Beyond Supervision, Seattle, US (2001)
7. Nakagawa, T., Matsumoto, Y.: Detecting errors in corpora using support vector machines. In: Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002), Taipei, TW, pp. 1–7 (2002)
8. Dickinson, M., Meurers, W.D.: Detecting errors in part-of-speech annotation. In: Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2003), Budapest, HU, pp. 107–114 (2003)
9. Fukumoto, F., Suzuki, Y.: Correcting category errors in text classification. In: Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004), Geneva, CH, pp. 868–874 (2004)
10. Argamon-Engelson, S., Dagan, I.: Committee-based sample selection for probabilistic classifiers. Journal of Artificial Intelligence Research 11, 335–360 (1999)

Semi-parametric and Non-parametric Term Weighting for Information Retrieval

Donald Metzler and Hugo Zaragoza

Yahoo! Research
{metzler,hugoz}@yahoo-inc.com

Abstract. Most of the previous research on term weighting for information retrieval has focused on developing specialized parametric term weighting functions. Examples include *TF.IDF* vector-space formulations, BM25, and language modeling weighting. Each of these term weighting functions takes on a specific parametric form. While these weighting functions have proven to be highly effective, they impose strict constraints on the functional form of the term weights. Such constraints may possibly degrade retrieval effectiveness. In this paper we propose two new classes of term weighting schemes that we call semi-parametric and non-parametric weighting. These weighting schemes make fewer assumptions about the underlying term weights and allow the data to speak for itself. We argue that these robust weighting schemes have the potential to be significantly more effective compared to existing parametric schemes, especially with the growing amount of training data becoming available.

1 Introduction

A great deal of research has been devoted to developing highly effective term weighting schemes for information retrieval. Some examples include *tf.idf* [1], pivoted length normalization [2], BM25 [3], language modeling [4], divergence from randomness [5], axiomatic weighting [6], genetic programming [7], and impact-based weighting [8]. Despite their differences, all of these approaches share one thing in common – they all assume that the underlying term weighting function takes on a specific functional form. Therefore, most, if not all, of the previously proposed term weighting schemes for information retrieval can be considered *parametric*.

Parametric term weighting functions restrict expressiveness and, possibly, effectiveness because the resulting weights are biased, *a priori*, to conform to the chosen functional form. Indeed, there is evidence that term weighting functions with more degrees of freedom, and therefore fewer functional restrictions, are more effective than weighting functions with fewer degrees of freedom. One classical example is that a well-tuned BM25, which has two parameters ($k1$, b) typically outperforms language modeling with Dirichlet smoothing, which has only just one parameter (μ). Of course, it is difficult to prove that the improved effectiveness is due to the extra degree of freedom, but it is certainly a possibility.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 42–53, 2009.

The current state-of-the-art term weighting schemes were developed when collections were small and training data was sparse. However, things are quite different now. Collections are larger than ever and training data is abundant, in the form of human judgments and click logs. We either have reached, or will soon reach, the point where we allow the data to "speak for itself", thereby eliminating the need to resort to parametric term weighting functions. While there has been some recent interest in developing *parameter-free* weighting functions [5], we believe that such models are better suited for "cold start" retrieval systems that have no training data, and that richer models with multiple parameters will be significantly more effective when training data is available.

In this paper, we look beyond traditional parametric term weighting functions, to more expressive weighting functions that have fewer functional constraints. Our primary contribution is two classes of term weighting functions that we call *semi-parametric* and *non-parametric* functions. As we will show, our proposed weighting functions mark a significant departure from previous term weighting research. We hypothesize this new direction could result in significant improvements in retrieval effectiveness and promote renewed interest in term weighting research.

The remainder of this paper is laid out as follows. First, in Section 2 we survey previous term weighting research. In Section 3 we describe our semi-parametric and non-parametric term weighting frameworks. Then, in Section 4 we discuss how the parameters of the proposed models can be estimated. Finally, we conclude the paper in Section 5 and describe possible avenues for future work.

2 Related Work

We now briefly describe three popular existing term weighting schemes. The first two, BM25 and language modeling, are based on probabilistic retrieval models. Such models are inherently parametric, because each assumes terms are distributed according to some parametric statistical model, such as a multinomial or mixture of Poissons. The other term weighting scheme that we discuss, which is based on ordinal term weighting, makes fewer assumptions about the data, although the weights still take on a parametric form. We will show how we can easily combine, and build upon each of these to develop even more robust weighting schemes.

2.1 BM25 Term Weighting

The classical probabilistic retrieval model ranks documents in decreasing order of likelihood of relevance, in accordance with the Probability Ranking Principle [9]. The general form of the model is:

$$S(Q,D) = P(r|q,d) \stackrel{rank}{=} \sum_{t \in Q \cap D} \log \frac{P(t|r)P(\overline{t}|\overline{r})}{P(\overline{t}|r)P(t|\overline{r})}$$

where t and $\overline{t}$ represent the occurrence and non-occurrence of term t, respectively. Furthermore, $P(t|r)$ and $P(t|\overline{r})$ represent the likelihood of event t in the relevant and non-relevant classes of documents, respectively. Previous researchers have made various distributional assumptions for these distributions. Assuming a multivariate-Bernoulli results in the Binary Independence Retrieval model [10], whereas the assumption of a 2-Poisson model, after some simplifying assumptions, results in the BM25 model [3]. The BM25 ranking function has the following form:

$$S(Q, D) = \sum_{t \in Q \cap D} \frac{tf_{t,D}}{k_1(1 - b + b\frac{|D|}{|D|_{avg}}) + tf_{t,D}} idf_t$$

where k_1 and b are free parameters that control for term frequency saturation and document length normalization.

2.2 Language Modeling Term Weighting

The language modeling framework for information retrieval is another widely used probabilistic model [4]. It is based on the assumption that documents can be topically represented by a probabilistic model called a *document model*. Document models are commonly modeled as multinomial distributions that are smoothed against the collection model in various ways [11]. Queries are similarly represented as *query models*, which are also typically modeled as multinomial distributions. Query models can be estimated in various ways, including maximum likelihood estimation, local blind feedback [12,13], or global blind feedback [14].

Documents are ranked according to the similarity between the query and document models. Kullback-Leibler divergence is often used as the (dis)similarity measure. Therefore, documents are ranked according to:

$$\begin{aligned} S(Q, D) &= -KL(P(\cdot|Q), P(\cdot|D)) \\ &\overset{rank}{=} \sum_{t \in \mathcal{V}} P(t|Q) \log P(t|D) \end{aligned}$$

where $P(\cdot|Q)$ and $P(\cdot|D)$ are the query and document models, respectively. Although the sum goes over the entire vocabulary $\mathcal{V}$, most query models are sparse, which significantly reduces the computational complexity. Language modeling term weights are parametric, where the parameterization depends on the type of smoothing used.

2.3 Ordinal Term Weighting

Document-centric impacts, originally proposed by Anh and Moffat [15], assign weights to document and query terms based on their relative importance to other terms. Terms are weighted as follows. First, given document, a total ordering of the (unique) terms is imposed. This is typically done by sorting the terms according to their term frequency and breaking ties with inverse document frequency. Once the terms have been totally ordered, they are partitioned into k bins. Here, it is assumed that all terms within bin k are equally important and that terms in bin $i + 1$ are more important than those in bin i. Essentially, the total ordering is transformed into a partial ordering. This binning procedure is typically done by geometrically binning the terms, where a small number of terms are considered "most important" (i.e., assigned to bin k) and a large number are considered "least important" (i.e., assigned to bin 1). A similar, yet slightly different, procedure is done on the query side to map query terms to bins.

Once terms have been assigned to bins, they must be assigned a weight for the purpose of scoring. Anh and Moffat, for simplicity, assign integral weights to the bins, with all of the terms within bin i being assigned weight i. We denote the weight for term w in documents and queries as $w_{bin(t,Q)}$ and $w_{bin(t,D)}$, respectively. Given a query Q and a document D, the score assigned under the Anh and Moffat model is:

$$\begin{aligned} S(Q,D) &= \sum_{w \in Q \cap D} w_{bin(t,Q)} w_{bin(t,D)} \\ &= \sum_{w \in Q \cap D} bin(t,Q) bin(t,D) \end{aligned}$$

where $bin(t,Q)$ and $bin(t,D)$ is the bin that term w is assigned in the query and document, respectively, and the equivalence follows from the fact that integral weights are used (i.e., $w_{bin(t,Q)} = bin(t,Q)$).

Anh and Moffat show that a very small number of bins is sufficient to achieve good retrieval effectiveness, but not as good as BM25 or language modeling. Adding more bins tends not to significantly improve effectiveness. Furthermore, fewer bins results in smaller indexes and considerably faster query execution times. Therefore, 4, 8, or 16 bins are often used in practice. One reason why the method tends to have sub-standard retrieval effectiveness compared to BM25 and language modeling is because of the choice of integral weights, which is an oversimplification. It has been shown that automatically learning the weights can lead to improvements in retrieval effectiveness [16].

3 Term Weighting

The term weighting problem for information retrieval requires a system to assign weights to word/query and word/document pairs. The weight should accurately reflect the importance of the term with respect to the query or document, with higher weights indicating more important terms. Of course, the ultimate goal is to assign term weights in such a way that the underlying retrieval model is highly effective according to some metric.

More formally, given a vocabulary $\mathcal{V}$, a set of documents $\mathcal{D}$, and a set of queries $\mathcal{Q}$ the term weighting problem requires us estimate $W \in \mathbb{R}^{|\mathcal{V}|} \times \mathbb{R}^{|\mathcal{Q}|} \times \mathbb{R}^{|\mathcal{D}|}$, where entry $w_{t,Q,D}$ corresponds to the weight of term t assigned to document D for query Q. We may also wish to condition the term weights on the user, time, or various other factors. However, for simplicity, we ignore these factors, as they would only complicate things and make our problem even more difficult to solve.

Our goal is to find the W that, when used in conjunction with the underlying (yet to be specified) ranking function, optimizes some evaluation metric of interest.

Some care must be taken when solving this problem, because there are a total of $|\mathcal{V}||\mathcal{D}||\mathcal{Q}|$ parameters. Obviously this estimation problem is infeasibly large for any non-trivial search task. This is one reason why parametric term weighting schemes have been so popular and appealing. Such schemes effectively reduce this enormous solution space down to just a handful of parameters.

In this paper, we assume that the underlying ranking function has the following form:

$$S(Q, D) = \sum_{t \in Q} w_{t,Q,D}$$

where Q is a query, D is a document, and $w_{t,Q,D}$ is the weight of t with respect to Q and D. We refer to this as the *joint form*, since the weight $w_{t,Q,D}$ depends jointly on Q and D.

While the joint formulation is the most generic way of representing most ranking functions, a vast majority of the widely used retrieval models, including BM25 and language modeling, can be written in a slightly simpler form, as follows:

$$S(Q, D) = \sum_{t \in Q} w_{t,Q} w_{t,D}$$

We refer to this as the *factored form*, since the weight $w_{t,Q,D}$ can be factored into the product of a weight for t in Q ($w_{t,Q}$) and a weight for t in D ($w_{t,D}$). This factorization reduces the size of the parameter space from $|\mathcal{V}||\mathcal{D}||\mathcal{Q}|$ to $|\mathcal{V}|(|\mathcal{D}| + |\mathcal{Q}|)$, which unfortunately is still infeasibly large.

Solving the term estimation problem, in either the joint or factored form, is simply not possible. Therefore, we must resort to measures that reduce the dimensionality of the problem while still maintaining expressiveness and effectiveness. We will now describe a general framework for reducing the term weighting dimensionality. We will then show how, within this framework, it is possible to develop whole new classes of term weighting schemes that make far fewer assumptions about the data than current approaches.

3.1 Dimensionality Reduction

There are various ways to reduce the dimensionality of the term weighting problem. Previous researchers have used latent semantic analysis [17,18], topic modeling [19], and various parametric functions (see Section 2) for this purpose. Rather than subscribe to any one of these approaches, we present the dimensionality reduction problem more generally, since we believe that information retrieval specific techniques may be superior to the previously proposed approaches.

Our proposed dimensionality reduction framework is very similar in spirit to the binning strategies used by Anh and Moffat [8]. In fact, their binning strategies can be used directly within our framework. However, as we will show, our framework is more general and results in a more formal estimation procedure.

When scoring a document D with respect to a query Q, we first bin the terms in the query and then bin the terms in the document. This binning can be thought of as a form of dimensionality reduction or massive parameter tying. We assume that the query terms are mapped (deterministically) into k_Q bins and document terms are mapped (deterministically) into k_D bins, where k_Q and k_D are fixed *a priori* and are constant across all queries and documents. Given the binned query terms and binned document terms, the retrieval status value (i.e., score) is computed as follows:

$$S(Q, D) = \sum_{t \in Q} \hat{w}_{bin(t,Q),bin(t,D)}$$

where $bin(t, Q)$ and $bin(t, D)$ are the query and document bins, respectively for term t, and $\hat{w}_{i,j}$ is an entry in $\hat{W} \in \mathbb{R}^{k_Q} \times \mathbb{R}^{k_D}$, which is a lower dimensional approximation of the full weight specification W. This approximation has $k_Q k_D$ parameters which is substantially smaller than both $|\mathcal{V}||\mathcal{D}||\mathcal{Q}|$ and $|\mathcal{V}|(|\mathcal{D}|+|\mathcal{Q}|)$.

It is important to note that, although binning and weighting are done on the term-level, the resulting models will not necessarily be *bag of words* models. The binning strategies may use contextual information, such as surrounding words, formatting, document structure, etc. Therefore, unlike traditional bag of words models, where a random permutation of the terms will result in the same term weights, our framework provides a simple mechanism for contextual term weighting.

Additionally, it should be clear that Anh and Moffat's model is a special case of this model, where $w_{bin(t,Q),bin(t,D)} = w_{bin(t,Q)} w_{bin(t,D)}$ and binning is done according to their proposed methods. However, as we will soon show, this dimensionality reduction framework can be applied to term weighting in a variety of interesting ways.

In order to use our proposed term weighting scheme we must define a query term binning strategy, a document term binning strategy, and a weighting $\hat{W}$. We will now describe the details of each of these steps.

Query Term Binning. There are various ways of perform query-side binning, including:

- Anh and Moffat query binning, which bins the query terms into $|Q|$ bins. The query term with the largest IDF is assigned to bin $|Q|$, the term with the next largest IDF is assigned to bin $|Q| - 1$, and so on, with the term with the smallest IDF being assigned to bin 1.
- Query-independent IDF binning. Rather than sorting terms within the query, we can bin terms according to their IDF. For example, we can assign the 25% of terms with the largest IDF out of the entire vocabulary to bin 4, the terms with the 25% next largest IDF to bin 3, and so on, with the 25% of terms with the lowest IDF to bin 1. There may be other ways of performing this binning, based on the number of documents that each term occurs in.
- Lexical binning. One may also use lexical information to assign words to bins in different ways. For example, some frequent and important words may be assigned their own bin, or bins could be assigned to types of words based on their length, their part of speech, their lexical semantics, etc.

Document Term Binning. Furthermore, several possible ways to bin document terms are:

- Anh and Moffat document binning, as described in Section 2.3.
- Binning based on existing term weighting schemes. For example, one can sort all of the terms within a document according to BM25, then assign terms to bins geometrically or uniformly. This is similar to the Anh and Moffat approach, except sorting is done in a slightly different manner.
- Lexical binning. As described in the query term binning section, we may assign terms to bins based on linguistic properties of the term.

3.2 Non-parametric Term Weighting

After query and document terms have been assigned bins, the final step is to determine the weightings $w_{bin(t,Q),bin(t,D)}$ for each combination of bins. As we described before, this problem has $k_Q k_D$ parameters. Depending on the number of bins, it may actually be possible to learn the term weighting directly, without the need to impose any parameterized form on the problem. When parameters are estimated in this way, we call the resulting weighting *non-parametric*, since the weights are learned directly from the data and have no pre-determined functional form.

Figure 1 summarizes the non-parametric term weighting problem. In this example, there are 3 query term bins ($k_Q = 3$) and 4 document term bins ($k_D = 4$). This results in a total of 12 parameters that can be directly estimated. We will describe methods for solving this estimation problem in Section 4.1.

The benefit of such a term weighting scheme is that it assumes no functional form for the term weights and learns directly from the data. However, this method relies on having very reliable, high quality query and document term binning functions. It may be the case that many bins will be necessary to accurately represent the importance of terms within queries and documents, potentially resulting in too many parameters to reliably estimate. The optimal binning strategy and number of bins is an open question that requires further investigation.

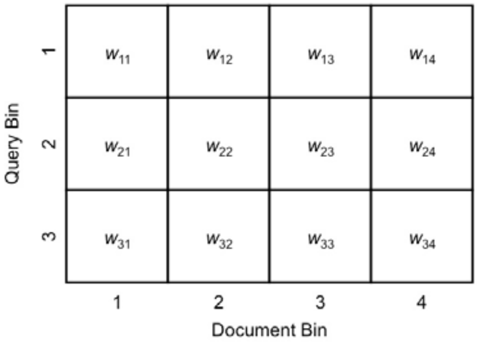


Fig. 1. Summary of the non-parametric term weighting problem after dimensionality reduction

3.3 Semi-parametric Term Weighting

In non-parametric term weighting, no functional form was assumed for the weights. As we discussed, depending on the binning strategies applied, this may result in too many parameters. One way of combating this issue is to solve a more constrained version of the term weighting problem that assumes some functional form for $\hat{w}_{bin(t,Q),bin(t,D)}$ but has parameters that depend on $bin(t,D)$ and $bin(t,D)$. We call this class of term weighting schemes *semi-parametric*, since the weighting function takes on a parametric form, but the parameters are not fixed across all term, query, document pairs, as in traditional parametric models. This scheme allows different classes of query and document terms to be weighted vastly differently, based on their characteristics.

As a concrete example, let us consider a semi-parametric form of BM25 weighting, which we propose as follows:

$$\hat{w}_{bin(t,Q),bin(t,D)} = \frac{tf_{t,D}}{k_{bin(t,Q)}(1 - b_{bin(t,D)} + b_{bin(t,D)}\frac{|D|}{|D|_{avg}}) + tf_{t,D}} idf_t$$

Here, it is important to notice that the term frequency saturation parameter k depends on the $bin(t,Q)$ and the document length normalization parameter b depends on $bin(t,D)$. In this way, we can model the fact that different types of terms tend to saturate in importance differently than others. For example, it may be that a single occurrence of a named entity is enough to saturate the term frequency, whereas many occurrences of a medium-IDF term may be required. Similarly, $bin(t,D)$ may be defined to be term independent and simply act to partition documents along some dimension, such as their length. In this way, we could have a document length-dependent setting for b.

A similar type of idea could be applied within the language modeling framework. For example, the following semi-parametric version of Dirichlet smoothing could be used:

$$\hat{w}_{bin(t,Q),bin(t,D)} = \alpha_{bin(t,Q)} \log \frac{tf_{t,D} + \mu_{bin(t,D)} P(t|C)}{|D| + \mu_{bin(t,D)}}$$

Within this formulation we have a different smoothing parameter μ for every bin $bin(t,D)$. This could allows us to use a different smoothing parameter for different classes of document lengths (e.g., short, medium, long), in a similar manner to the semi-parametric b just proposed to be used in conjunction with BM25. We also define a parameter α that depends on $bin(t,Q)$. This can be used to weight different classes of query terms differently. For example, we may want to upweight nouns and downweight certain adjectives, definitives, etc. This can all be accomplished by defining appropriate query and document term binning strategies.

It may be possible to learn more generic weighting functions in this way, as well. For example, a linear or non-linear regression model may be used as the parametric form, with the model parameters depending on $bin(t,Q)$ and $bin(t,D)$. Similar semi-parametric forms can be used to estimate the weight of a term in the query or even a joint weight that depends on both the query and the document (i.e., $w_{t,D,Q}$).

3.4 Parametric Term Weighting

It should now be clear that standard parametric term weighting functions are special cases of our framework that trivially assign all query and document terms to the same bin (i.e., $k_D = k_Q = 1$). Therefore, our framework can be used to expand the expressiveness of any existing term weighting scheme by providing a mechanism to use more fine-grained parameters, which we hypothesize will lead to improvements in retrieval effectiveness.

4 Estimating $\hat{W}$

4.1 Non-parametric Weight Estimation

The ideal situation is to estimate $\hat{W}$ in a supervised manner using training data in the form of human judgments or click data. Estimating $\hat{W}$ can be transformed into a standard linear "learning to rank" problem. It can be shown that $S(Q, D)$, as defined above, can be rewritten as:

$$S(Q,D) = \sum_{i=1}^{k_Q} \sum_{j=1}^{k_D} |\{w \in Q : bin(t,Q) = i, bin(t,D) = j\}| \hat{w}_{i,j}$$

which is a linear function with respect to the weights $\hat{w}_{i,j}$. If we treat the $|\{w \in Q : bin(t,Q) = i, bin(t,D) = j\}|$ values as "features", then this is a standard linear learning to rank problem, by which we want to find the weights $\hat{w}_{i,j}$ that optimize for some evaluation metric, such as mean average precision or NDCG. A variety of techniques have been described for solving this problem [20,21].

The weights $\hat{w}_{i,j}$ learned as the result of this optimization process are then used for scoring. It is important to note that while $S(Q, D)$ is parametric (i.e., linear), the term weights $\hat{w}_{i,j}$ are not, since they may take on any possible value. Finally, we point out that although the scoring function is linear with respect to $\hat{w}_{i,j}$, the weights may be non-linear with respect to term frequency and inverse document frequency, depending on the binning strategy.

If little or no training data is available, then we can use existing term weighting functions, such as BM25, to estimate $\hat{W}$, as follows:

$$\hat{w}_{i,j} = \frac{\sum_{Q \in \mathcal{Q}} \sum_{D \in \mathcal{D}} \sum_{w \in Q \cap D : bin(t,Q)=i, bin(t,D)=j} BM25(t, D)}{\sum_{Q \in \mathcal{Q}} \sum_{D \in \mathcal{D}} |\{w \in Q \cap D : bin(t,Q) = i, bin(t,D) = j\}|}$$

where $\mathcal{Q}$ and $\mathcal{D}$ is the universe of queries and documents, respectively. This unsupervised estimate simply averages (over the entire universe of queries and document) the BM25 term weights for each entry in $\hat{W}$. Of course, it is infeasible to compute this value exactly since $|\mathcal{Q}||\mathcal{D}|$ is essentially unbounded. Instead, a reasonable estimate can be derived by sampling a small number queries and documents to compute the average over. This unsupervised estimate can also be used as a prior, or starting point, when estimating $\hat{W}$ in a supervised manner.

4.2 Parametric and Semi-parametric Weight Estimation

Parameter estimation for parametric and semi-parametric term weighting schemes is slightly more difficult, since the functional forms are likely to be non-linear with respect to the parameters. This may pose a challenge when using standard optimization techniques. It may be possible to use an approach, such as the one described by Taylor et al. to optimize a proxy loss function, as long as the weight function is differentiable with respect to the parameters [22].

Depending on the complexity of the underlying parametric form, the evaluation metric of interest, the size of the test collection, and the number of parameters, a simple grid search or greedy search technique may work just fine. However, if the number of parameters is large, as will be the case when k_D and/or k_Q is large, then more careful optimization procedures must be devised to avoid possible overfitting and excessive computational costs.

5 Conclusions and Future Work

This paper described a spectrum of term weighting schemes for information retrieval that go beyond traditional parametric weighting. In particular we proposed semi-parametric and non-parametric weighting schemes that we hypothesize could result in more robust, more effective retrieval models. Table 1 provides a summary of the different weighting schemes that were discussed.

As we showed, *non-parametric* weighting schemes are the most generic of the three types. These weighting schemes do not assume any functional form for the weights. Instead, the weights are estimated directly. While this is the most general approach, we suspect that a large number of parameters may be necessary to provide good retrieval effectiveness in practice, and therefore a very large amount of training data may be necessary to effectively learn such models. However, if such data is available, we suspect that these models have the potential to yield significant improvements in retrieval effectiveness.

The next most generic class of weighting schemes are *semi-parametric*. Under this scheme, weighting functions have some parametric form, but the parameters of the weighting functions depend on the query term and document term binnings. In this way, the weights are parametric, but depending on the binning, can be adapted better to the data due to the less constrained parameterization.

Finally, *parametric* weighting schemes, which account for most, if not all, of the currently used term weighting functions are the most restrictive. In this class, weighting functions are parametric, but the parameters (if any) of the

Table 1. Summary of the different types of term weighting schemes, their functional form, and their parameters

Type of Weighting	Functional Form	Parameters
Parametric	Parametric Function	Global
Semi-Parametric	Parametric Function	Dependent on w, Q, D
Non-parametric	No Functional Form	Dependent on w, Q, D

weighting scheme are global. That is, the same set of parameters are applied to all queries and documents. While the global parameters are estimated from data, the underlying weights may not be very adaptable to a wide variety of query terms and document types, thereby hindering effectiveness.

This paper was devoted entirely to the theory underlying different classes of term weighting functions. However, an important direction of future work is to understand the implications of the theory on practical information retrieval systems. In particular, we plan to explore the effectiveness of the different classes of term weighting schemes. We also plan to develop a better grasp on the usefulness of different binning strategies and the feasibility of using completely non-parametric term weighting.

References

1. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing and Management 24(5), 513–523 (1988)
2. Singhal, A., Buckley, C., Mitra, M.: Pivoted document length normalization. In: Proc. 19th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 21–29 (1996)
3. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: Proc. 17th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 232–241. Springer, New York (1994)
4. Ponte, J., Croft, W.B.: A language modeling approach to information retrieval. In: Proc. 21st Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 275–281 (1998)
5. Amati, G., van Rijsbergen, C.J.: Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Transactions on Information Systems 20(4), 357–389 (2002)
6. Fang, H., Zhai, C.: An exploration of axiomatic approaches to information retrieval. In: Proc. 28th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 480–487 (2005)
7. Fan, W., Gordon, M.D., Pathak, P.: A generic ranking function discovery framework by genetic programming for information retrieval. Inf. Process. Manage. 40(4), 587–602 (2004)
8. Anh, V.N., Moffat, A.: Simplified similarity scoring using term ranks. In: Proc. 28th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 226–233 (2005)
9. Robertson, S.: The probability ranking principle in IR. Journal of Documentation 33(4), 294–303 (1977)
10. Robertson, S.E., Spärck Jones, K.: Relevance weighting of search terms. Journal of the American Society for Information Science 27(3), 129–146 (1976)
11. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. In: ACM Trans. Inf. Syst., vol. 22(2), pp. 179–214 (2004)
12. Lavrenko, V., Croft, W.B.: Relevance-based language models. In: Proc. 24th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 120–127 (2001)

13. Zhai, C., Lafferty, J.: Model-based feedback in the language modeling approach to information retrieval. In: Proc. 10th Intl. Conf. on Information and Knowledge Management, pp. 403–410 (2001)
14. Berger, A., Lafferty, J.: Information retrieval as statistical translation. In: Proc. 22nd Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 222–229 (1999)
15. Anh, V.N., Moffat, A.: Collection-independent document-centric impacts. In: Proc. Australian Document Computing Symposium, pp. 25–32 (2004)
16. Metzler, D., Strohman, T., Croft, W.B.: A statistical view of binned retrieval models. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 175–186. Springer, Heidelberg (2008)
17. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. Journal of the American Society for Information Science 41, 391–407 (1990)
18. Hofmann, T.: Probabilistic latent semantic indexing. In: Proc. 22nd Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 50–57. ACM, New York (1999)
19. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022 (2003)
20. Gao, J., Qi, H., Xia, X., Nie, J.Y.: Linear discriminant model for information retrieval. In: Proc. 28th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, pp. 290–297. ACM, New York (2005)
21. Joachims, T.: A support vector method for multivariate performance measures. In: Proc. 22nd Proc. Intl. Conference on Machine Learning, pp. 377–384. ACM, New York (2005)
22. Taylor, M., Zaragoza, H., Craswell, N., Robertson, S., Burges, C.: Optimisation methods for ranking functions with multiple parameters. In: Proc. 15th Intl. Conf. on Information and Knowledge Management, pp. 585–593. ACM, New York (2006)

Bridging Language Modeling and Divergence from Randomness Models: A Log-Logistic Model for IR

Stéphane Clinchant[1,2] and Eric Gaussier[2]

[1] Xerox Research Center Europe, 6 chemin de Maupertuis 38240, Meylan France
stephane.clinchant@xrce.xerox.com
[2] LIG,Univ. Grenoble I, BP 53 - 38041 Grenoble cedex 9, Grenoble France
eric.gaussier@imag.fr

Abstract. We are interested in this paper in revisiting the Divergence from Randomness (DFR) approach to Information Retrieval (IR), so as to better understand the different contributions it relies on, and thus be able to simplify it. To do so, we first introduce an analytical characterization of heuristic retrieval constraints and review several DFR models wrt this characterization. This review shows that the first normalization principle of DFR is necessary to make the model compliant with retrieval constraints. We then show that the log-logistic distribution can be used to derive a simplified DFR model. Interestingly, this simplified model contains Language Models (LM) with Jelinek-Mercer smoothing. The relation we propose here is, to our knowledge, the first connection between the DFR and LM approaches. Lastly, we present experimental results obtained on several standard collections which validate the simplification and the model we propose.

1 Introduction

Together with the language modeling approach to IR, Divergence from Randomness (DFR) models, recently introduced by Amati and Van Rijsbergen [2], are among the best performing (and thus most used) IR models in international evaluation campaigns as TREC or CLEF. However, the DFR framework is complex and difficult to comprehend, as it relies on several quantities the role of which is not always clear. We are interested here in trying to better understand this framework so as to simplify it. Interestingly, the simplification we arrive at contains standard language models with Jelinek-Mercer smoothing.

The remainder of the paper is organized as follows: Section 3 introduces an analytical characterization of IR heuristics which will be used throughout the paper; Section 3 describes the DFR framework and lists the problems associated with it; Section 4 describes the simplification we propose on the basis of the log-logistic distribution, and the relation with language models; Section 5 finally presents an experimental validation of our simplification. Throughout the paper, we make use of the following notations: $\mathcal{C}$ is a collection of N documents; for each

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 54–65, 2009.

index term w, x_w^d (resp. x_w^q) will represent the number of occurrences of the term in document d (resp. in query q), n_w the number of documents in which the term occurs, F_w the number of occurrences of the term in the whole collection, and z_w a quantity which, depending on the context, is equal to n_w or F_w, potentially normalized by N (we introduce this quantity to simplify the expression of the different models we are going to consider); y_d will denote the length of document d, and avdl the average length of the documents in the collection.

2 Analytical Characterization of IR Heuristics

Following Fang *et al.* [6], who proposed formal definitions of heuristic retrieval constraints which can be used to assess the validity of an IR model, we introduce here analytical conditions a retrieval function should satisfy to be valid.

We consider here retrieval functions, denoted RSV, of the form:

$$RSV(q,d) = \sum_{w \in q \cap d} h(x_w^d, y_d, z_w, \theta)$$

where θ is a set of parameters and where h, the form of which depends on the IR model considered, is assumed to be of class C^2 and defined over $\mathbb{R}^{+*} \times \mathbb{R}^{+*} \times \mathbb{R}^{+*} \times \Theta$, where Θ represents the domain of the parameters in θ. The above form encompasses many IR models, as the vector space model, language models, or divergence from randomness models. For example, for the pivoted normalization retrieval formula [9], $\theta = (s, \text{avdl}, N, x_w^q)$ and:

$$h(x,y,z,\theta) = \frac{1+\ln(1+\ln(x))}{1-s+s\frac{y}{\text{avdl}}} x_w^q \ln(\frac{N+1}{z})$$

A certain number of hypotheses, experimentally validated, sustain the development of IR models. In particular, it is important that documents with more occurrences of query terms get higher scores than documents with less occurrences. However, the increase in the retrieval score should be smaller for larger term frequencies, inasmuch as the difference between say 110 and 111 is not as important as the one between 1 and 2 (the number of occurrences has doubled in the second case, whereas the increase is relatively marginal in the first case). In addition, longer documents, when compared to shorter ones with exactly the same number of occurrences of query terms, should be penalized as they are likely to cover additional topics than the ones present in the query. Lastly, it is important, when evaluating the retrieval score of a document, to weigh down terms occurring in many documents, i.e. which have a high document/collection frequency, as these terms have a lower discrimination power. We formalize these considerations through the following four conditions (a larger set of conditions as well as their relation to the formal definitions proposed by Fang *et al.* [6] are given in Appendix A):

Condition 1. $\forall (y,z,\theta),\ \dfrac{\partial h(x,y,z,\theta)}{\partial x} > 0$; **Condition 2** $\forall (y,z,\theta),\ \frac{\partial^2 h(x,y,z,\theta)}{\partial x^2} < 0$

Condition 3. $\forall (x,z,\theta), \dfrac{\partial h(x,y,z,\theta)}{\partial y} < 0$; **Condition 4** $\forall (x,y,\theta), \frac{\partial h(x,y,z,\theta)}{\partial z} < 0$

Conditions 1, 3 and 4 directly state that h should be increasing with the term frequency, and decreasing with the document length and the document/collection frequency. Conditions 1 and 2 (mentioned by Fang *et al.* [6]) state that h should be an increasing, concave function of the term frequency, the concavity ensuring that the increase in the retrieval score will be smaller for larger term frequencies.

3 The DFR Framework

The Divergence from Randomness (DFR) framework proposed by Amati and van Rijsbergen [2] is currently one of the most successful IR model. It is based on the informative content provided by the occurrences of terms in documents, a quantity which is then corrected by the risk of accepting a term as a descriptor in a document (*first normalization principle*) and by normalizing the raw occurrences by the length of a document (*second normalization principle*). In the remainder, $t(x_w^d, y_d)$ will denote the normalized form of x_w^d. The informative content $\mathit{Inf}_1(t(x_w^d, y_d))$ is based on a first probability distribution and is defined as: $\mathit{Inf}_1(t(x_w^d, y_d)) = -\log \mathit{Prob}_1(t(x_w^d, y_d))$. The first normalization principle is associated with a second information defined from a second probability distribution through: $\mathit{Inf}_2(t(x_w^d, y_d)) = 1 - \mathit{Prob}_2(t(x_w^d, y_d))$. For example, using the Laplace law of succession for the first normalization (Prob_2), one obtains the following retrieval function:

$$RSV(q,d) = \sum_{w \in q \cap d} x_w^q \overbrace{\left(\frac{1}{t(x_w^d, y_d) + 1} \right)}^{\mathrm{Inf}_2(t(x_w^d, y_d))} \mathrm{Inf}_1(t(x_w^d, y_d)) \qquad (1)$$

We now review the two normalization principles behind DFR models.

3.1 The Second Normalization Principle

The second normalization principle aims at normalizing the number of occurrences of words in documents by the document length, as a word is more likely to have more occurrences in a long document than in a short one. The different normalizations considered in the literature transform raw number of occurrences into positive real numbers. Language models for example use the relative frequency of words in the document and the collection. Other classical term normalization schemes include the well know Okapi normalization, as well as the pivoted length normalization [9]. More recently, [8] propose another formulation for the language model using the notion of verbosity.

DFR models usually adopt one of the two following term frequency normalizations (c is a multiplying factor):

$$t(x_w^d, y_d) = x_w^d c \frac{\text{avdl}}{y_d} \tag{2}$$

$$t(x_w^d, y_d) = x_w^d \log(1 + c\frac{\text{avdl}}{y_d}) \tag{3}$$

The important point about the second normalization principle is that, to be fully compliant with these definitions, the probability distribution functions at the basis of DFR models should be continuous distributions, which is not the case for the distributions usually retained in DFR models.

3.2 The First Normalization Principle

The intuition behind *Inf*$_1$ is simple. Let $P(t(x_w^d, y_d)|\lambda_w)$ represent the probability of $t(x_w^d, y_d)$ (normalized) occurrences of term w in document d according to parameters λ_w which are estimated or set on the basis of a random distribution of w in the collection. If $P(t(x_w^d, y_d)|\lambda_w)$ is low, then the distribution of w in d deviates from its distribution in the collection, and w is important to describe the content of d. In this case, *Inf*$_1$ will be high. On the contrary, if $P(x_w^d|\lambda_w)$ is high, then w behaves in d as expected from the whole collection and, thus, does not provide much information on d (*Inf*$_1$ is low). *Inf*$_1$ thus captures the importance of a term in a document through its deviation from an average behavior estimated on the whole collection. The question which thus arises is why one should need to normalize it. In other words, what is the role of the first normalization principle?

Amati and van Rijsbergen [2] consider five basic IR models for $Prob_1$: the binomial model, the Bose-Einstein model, which can be approximated by a geometric distribution, the *tf-idf* model (denoted *I(n)*), the tf-itf model (denoted *I(F)*) and the *tf-expected-idf* model (denoted *I(n$_e$)*). For the last four models, *Inf*$_1$ takes the form:

$$Inf_1(t(x_w^d, y_d)) = \begin{cases} t(x_w^d, y_d)\log(1 + \frac{N}{z_w}) + \log(1 + \frac{z_w}{N}) \\ t(x_w^d, y_d)\log(\frac{N+1}{z_w+0.5}) \end{cases}$$

where the first line corresponds to the geometric distribution, and the second one to *I(n)*, *I(F)* and *I(n$_e$)* (z_w being respectively equal to n_w, F_w and $n_{w,e}$, the latter representing the expected number of documents containing term w). We assume in the remainder that $t(x_w^d, y_d)$ is given either by equation 2 or 3. The conclusions we present below are the same in both cases.

Were we to base a retrieval function on the above formulation of *Inf*$_1$ only, our model would be defined by:

$$\theta = (x_w^q, \text{avdl}, N)$$

$$h(x, y, z, \theta) = \begin{cases} x_w^q \left(t(x, y)\log(1 + \frac{N}{z}) + \log(1 + \frac{z}{N})\right) \\ x_w^q \left(t(x, y)\log(\frac{N+1}{z+0.5})\right) \end{cases}$$

where the first line still corresponds to the geometric distribution, and the second one to *I(n)*, *I(F)* and *I(n_e)*. It is straightforward to see that models *I(n)*, *I(F)* and *I(n_e)* verify conditions 1, 3 and 4 and that the model for the geometric distribution verifies conditions 1 and 3, but only partly condition 4, as the derivative can be positive for some values of z, N and t. All models however fail condition 2 as, in all cases, $\frac{\partial^2 h(x,y,z,\theta)}{\partial x^2} = 0$. Hence, Inf_1 alone, for the geometric distribution and the models *I(n)*, *I(F)* and *I(n_e)*, is not sufficient to define a valid IR model[1]. One can thus wonder whether Inf_2 serves to make the model compliant with condition 2. We are going to see that this is indeed the case.

Two quantities are usually used for Inf_2 in DFR models: the normalization L or the normalization B. They both lead to the following form:

$$\mathrm{Inf}_2 = \frac{a}{t(x_w^d, y_d) + 1}$$

where a is independent of $t(x_w^d, y_d)$. Thus integrating Inf_2 in the previous models gives:

$$h(x,y,z,\theta) = \begin{cases} x_w^q \left(\frac{at(x,y)}{t(x,y)+1} \log(1 + \frac{N}{z}) + \log(1 + \frac{z}{N}) \right) \\ x_w^q \left(\frac{at(x,y)}{t(x,y)+1} \log(\frac{N+1}{z+0.5}) \right) \end{cases}$$

As $\frac{\partial^2 h(x,y,z,\theta)}{\partial x^2} = \frac{\partial^2 h(x,y,z,\theta)}{\partial t^2} \left(\frac{\partial t}{\partial x}\right)^2$, and $\left(\frac{\partial t}{\partial x}\right)^2 > 0$ for the normalizations considered (equations 2 and 3), we have:

$$\mathrm{sgn}\left(\frac{\partial^2 h(x,y,z,\theta)}{\partial x^2}\right) = \mathrm{sgn}\left(\frac{\partial^2 h(x,y,z,\theta)}{\partial t^2}\right)$$

But:

$$\frac{\partial^2 h(x,y,z,\theta)}{\partial t^2} = -\frac{b}{(t(x_w^d, y_d) + 1)^3}$$

with $b > 0$, which shows that the models are now compatible with condition 2.

4 A Simplified DFR Approach to IR

Clinchant and Gaussier [4] propose to use a model relying solely on Inf_1, for which they retain the Beta negative binomial (BNB) distribution. The BNB distribution is based on the negative binomial distribution which has been proposed as an alternative to the standard Poisson or binomial models traditionnaly used in IR ([3,1,5]). The negative binomial is an infinite mixture of Poisson distributions, and thus can be considered as an extension of the Two-Poisson model. In [4], the Beta BNB distribution is introduced by using a uniform Beta prior on one of the negative binomial parameters. As those distributions are conjugate, the BNB results in a simple form:

[1] The same applies to the binomial model, for which $\frac{\partial^2 h(x,y,z,\theta)}{\partial x^2} > 0$. For the sake of clarity, we do not present here this derivation which is purely technical.

$$P_{BNB}(x_w^d|r_w) = \frac{r_w}{(r_w + x_w^d)(r_w + x_w^d + 1)}$$

where r_w^d is a parameter which can either be learned through maximum likelihood, or directly set from the collection (Clinchant and Gaussier suggest to use $\frac{F_w}{N}$, i.e. z_w with our notation). Using this distribution leads to the following IR model:

$$h(x, y, z, \theta) = -x_w^q \log(z) + x_w^q \log((z + t(x, y))(z + t(x, y) + 1))$$

With both length normalizations we consider here, it is easy to see that the above model verifies conditions 1, 2 and 3. However, condition 4 is only partly verified as, using for example equation 3, one obtains: $\mathrm{sgn}(\frac{\partial^2 h(x,y,z,\theta)}{\partial x^2}) = \mathrm{sgn}(z - x\log(1 + c\frac{\mathrm{avdl}}{y}))$, a quantity which can be negative for words appearing a lot of times in the collection (z high) but not often (x low) in a long document (y high). In addition, the above model still suffers from the fact that the underlying distribution is discrete, and yet applied to positive, real-valued variables. We introduce now a new distribution which corrects this problem.

4.1 A Log-Logistic Model for IR

There exists a distribution which can be seen, under certain conditions, as a continuous approximation of the BNB, namely the log-logistic distribution. The log-logistic distribution is a continuous distribution defined on the set of positive real numbers. Its density and cumulative probability function have the form[2].:

$$f_{LL}(x|\alpha, \beta) = \frac{(\beta/\alpha)(x/\alpha)^{\beta-1}}{(1 + (x/\alpha)^\beta)^2} \quad P_{LL}(X < x|\alpha, \beta) = \frac{x^\beta}{x^\beta + \alpha^\beta}$$

Setting $\alpha = r$ and $\beta = 1$ in the log-logistic distribution, we have:

$$\forall x \in \mathbb{R}^+,\ P_{LL}(x \leq X < x+1|r) = \frac{x+1}{r+x+1} - \frac{x}{r+x} = \frac{r}{(r+x+1)(r+x)} \quad (4)$$

Therefore, for all integer n, $P_{LL}(n \leq X < n+1|r) = P_{BNB}(X = n|r)$ and $P_{LL}(X > n|r) = P_{BNB}(X > n|r)$, which explains why the log-logistic distribution can be considered as a continuous approximation of the BNB distribution.

A simplified DFR model based on *Inf*$_1$ only and the log-logistic distribution can thus be defined by:

$$RSV(q, d) = \sum_{w \in q \cap d} -x_w^q \log(P_{LL}(X \geq t(x_w^d, y_d)|r_w))$$

where r_w is defined from the whole collection. In the remainder, we consider that r_w is set to either $\frac{F_w}{N}$ or $\frac{n_w}{N}$, a standard setting for the parameter of DFR models.

[2] For more information on the Log Logistic distribution refer to http://en.wikipedia.org/wiki/Log-logistic_distribution

The above ranking function corresponds to the mean information a document brings to a query (or, equivalently, to the average of the document information brought by each query term). Using the notation of previous sections, the IR model thus defined corresponds to:

$$h(x, y, z, \theta) = x_w^q \log(\frac{z + t(x, y)}{z})$$

This time, this model verifies conditions 1, 2, 3 and 4, for all the admissible values of x, y and z. It can also be shown that it verifies the other conditions associated with IR heuristic constraints and given in Property 3 of Appendix A.

We are thus now armed with a simplified DFR model, relying solely on Inf_1, which is compatible with the theoretical framework we have developed: our model is based on a continuous distribution that verifies the conditions of retrieval heuristic constraints. We now need to experimentally validate the fact that this model behaves as more complex DFR models on IR collections. We will do that in section 5. Prior to that, we want to show a connection between our model and the language modeling approach to IR.

4.2 Relation to Language Models

Let L be the number of tokens in the collection. Following [10], the scoring formula for a language model using Jelinek-Mercer smoothing can be written as:

$$RSV(q, d) = \sum_{w \in q \cap d} x_w^q \log(1 + s\frac{\frac{x_w^d}{y_d}}{\frac{F_w}{L}}) \tag{5}$$

Using the log-logistic model introduced previously with $r_w = \frac{F}{N}$ and the length normalization given by equation 2, we have:

$$RSV(q, d) = \sum_{w \in q \cap d} x_w^q \log(1 + c\frac{\frac{x_w^d \times \text{avdl}}{y_d}}{\frac{F_w}{N}}) \tag{6}$$

Given that $\frac{F_w}{N} = \text{avdl} \times \frac{F_w}{L}$, equation 5 is equivalent to equation 6. The LM model with Jelinek-Mercer smoothing can thus be seen as a log-logistic model with a particular length normalization, namely the one given by equation 2.

In the language modeling approach to IR, one starts from term distributions estimated as the document level, and smoothed by the distribution at the collection level. In contrast, the DFR approach uses a distribution the parameters of which are estimated on the whole collection to get a local document weight for each term. Despite the different views sustaining these two approaches, the above development shows that they can be reconciled through appropriate word distributions, in particular the log-logistic one. The DFR framework, and its simplification introduced here, is in a sense more general than the language modeling approach to IR since several length normalizations and several distributions can be considered, leading to a class of model encompassing the language modeling one, as shown above. Lastly, the above connection also indicates that term frequency or length normalizations are related to smoothing. A theory for relating these two elements remains however to be established.

5 Experimental Validation

We use the following collections to assess the validity of our model: TREC-3, ROBUST (TREC), CLEF03 AdHoc Task, GIRT (CLEF Domain Specific2004-2006), TEL British Library (CLEF'08 AdHoc). Table 1 gives the number of documents (N), number of unique terms (M), average document length and number of test queries for these collections. For the TREC-3, ROBUST and NTCIR collections, we used standard Porter stemming. For the CLEF03, GIRT and TEL collections, we used lemmatization, and an additional decoumpounding step for the GIRT collection which is written in German.

5.1 IR Results

As our model is a simplification of the DFR framework making use of a single distribution, the log-logistic one, we want to show experimentally that this simplification does not degrade IR results. The methodology we follow for that is straightforward: compute the mean average precision (MAP) for DFR models and the log-logistic one on several IR collections, and assess whether the difference in the MAP is significant or not.

We used three variants of the log-logistic model: a discrete variant based on the BNB distribution and two direct models, one with r_w set to the mean frequency of the word in the collection, $r_w = \frac{F_w}{N}$, model LG, and one with r_w set to the document frequency $\frac{n_w}{N}$, model LGD. These models were first tested against the standard PL2 and InL2 DFR models. In all cases, we used the length normalization corresponding to equation 3, and chose 4 different settings for c: $c = (0.5, 1, 5, 10)$, which corresponds to the typical range recommended for DFR models. Table 2 shows the MAP and precision at 10 for all the DFR and log-logistic models. TREC3-t refers to the TREC-3 collection with query title only, whereas TREC3-d uses both title and description fields (and similarly for the ROBUST collection with ROB-T and ROB-d). CLEF03 and GIRT queries are long queries with descriptive fields, whereas TEL-BL queries are typically short

Table 1. Characteristics of the different collections

	N	M	Avg DL	# Queries
TREC-3	741 856	668 482	262	50
ROBUST	490 779	992 462	289	250
CLEF03	166 754	80 000	247	60
GIRT	151 319	179 283	109	75
TEL-BL	870 246	259 569	9	50

Table 2. Mean average precision (MAP) and Precision at 10, for the different models and datasets. Bold indicates best performance per line (c value); $*$ indicates a significative difference with the best result by line at the 0.05 level for both T-test and Wilcoxon, whereas † indicates a significant difference with one test only.

		MAP					P10				
	c	BNB	LG	LGD	PL2	INL2	BNB	LG	LGD	PL2	INL2
TREC3-t	0.5	24.7	25.0^*	**25.3**	21.6^*	23.8^*	**50.0**	49.2	**50.0**	46.8	48.6
	1	25.6	25.7	**25.8**	24.3^*	25.5	51.8	52.6	**53.4**	51.0	51.0
	5	26.3^*	25.8^*	25.7^*	**27.0**	25.5^*	54.2	53.2	52.8	**55.2**	53.4
	10	25.6^*	25.3^*	25.0^*	**26.7**	24.8^*	53.6	52.2	51.0^*	**53.8**	$51.4^\dagger$
TREC3-d	0.5	28.4	**28.9**	**28.9**	25.8^*	28.4	**62.8**	59.8^*	58.0^*	57.0^*	59.8
	1	$28.5^\dagger$	28.6	27.9^*	28.7	**29.4**	62.4	58.6^*	58.2^*	**63.4**	60.4^*
	5	24.7^*	24.5^*	$23.0*$	**28.3**	25.5^*	54.6^*	53.6^*	51.6^*	**62.4**	54.2^*
	10	21.9^*	21.7^*	20.3^*	**26.0**	22.5^*	49.4^*	48.4^*	45.2^*	**54.2**	47.2^*
ROB-t	0.5	23.3^*	23.8^*	**24.0**	20.0^*	22.2^*	41.7	41.9	**42.0**	38.9^*	40.8
	1	23.8^*	24.2	**24.3**	22.2^*	$23.6^\dagger$	**42.6**	42.5	42.5	41.4	**42.6**
	5	24.3^*	**24.7**	**24.7**	**24.7**	24.5	$43.5^\dagger$	43.6	43.3^*	**44.5**	42.9^*
	10	24.1^*	$24.5^\dagger$	24.5	**24.8**	24.4	$43.4^\dagger$	43.8	43.7	**44.3**	43.0
ROB-d	0.5	26.6^*	27.1^*	**27.4**	23.0^*	25.5^*	**46.3**	45.5	45.6	43.5^*	46.0
	1	26.5	**26.9**	**26.9**	25.4^*	**26.9**	45.8	45.3^*	$45.6^\dagger$	$45.7^\dagger$	**47.0**
	5	24.5^*	24.6^*	24.2^*	**26.5**	25.5^*	44.3^*	42.9^*	42.0^*	**46.1**	42.8^*
	10	23.1^*	23.1^*	22.8^*	**25.6**	$23.8*$	41.2^*	40.7^*	40.1^*	**45.0**	40.2^*
CLEF03	0.5	47.8	48.8	**49.3**	44.0^*	47.3	34.3	**35.0**	34.8	$32.3^\dagger$	33.8
	1	47.7	48.4	48.2	46.2^*	**49.3**	33.6	34.3	**34.5**	33.5	34.33
	5	42.7^*	45.2^*	44.0^*	**47.0**	46.2	$32.2^\dagger$	$31.8^\dagger$	32.2	**33.8**	32.7
	10	39.5^*	41.0^*	39.7^*	**45.0**	43.8	$31.5^\dagger$	31.3	30.7^*	**32.8**	31.0
GIRT	0.5	40.2^*	40.4^*	**42.1**	35.0^*	39.8^*	**67.8**	67.1	67.5	62.5^*	66.5
	1	41.0^*	41.4^*	**42.3**	38.5^*	41.5	**69.5**	67.8^*	68.9	65.5^*	67.0^*
	5	41.3	41.7	41.6	**41.8**	40.5^*	**70.4**	68.4^*	68.7^*	69.7	66.1^*
	10	41.0^*	$41.2^\dagger$	41.2	**42.0**	39.7^*	**70.0**	68.0^*	67.8^*	69.6	65.2^*
TEL-BL	0.5	31.5^*	**33.0**	**33.0**	21.6^*	24.8^*	47.2	**47.8**	47.6	35.6^*	41.0^*
	1	31.6^*	32.5	**33.3**	26.8^*	29.5^*	47.6	48.4	**49.0**	43.0^*	45.8
	5	$32.0^\dagger$	32.5	33.0	31.3^*	**33.4**	49.8	49.4	**50.0**	$47.4^\dagger$	48.6
	10	31.8^*	32.4^*	32.9^*	31.8^*	**33.6**	49.0	49.6	50.0	48.8	**50.2**

queries, containing only few keywords. The PL2 model seems to give better results with $c = 5$, whereas the log-logistic ones tend to prefer $c = 0.5$ and InL2 $c = 1$. A two-sided T-test as well as a Wilcoxon test were computed between the results obtained with the best performing model and the results of other models with the same parameter setting. A $*$ in table 2 indicates that the difference between the two models considered is significant with both tests ($p = 0.05$), whereas a † indicates that the difference is significant with only one test (again with $p = 0.05$). For the MAP, out of 28 runs, log-logistic models perform slightly better than standard DFR models 13 times, and slightly worse 15 times. In most cases, there is no significant difference between the first and second best results (which are often obtained with two different models: standard DFR or log-logistic). Moreover, over all parameter settings, there is no significant difference between the best standard DFR models and the best log-logistic models on all collections but TREC3-t, for which the PL2 model outperforms the other ones. For the precision at 10 documents, log-logistic models are slightly better 15 times out of 28 (and slightly worse 13 times). Over all parameter settings, there is no significant difference between standard DFR models and log-logistic ones on all collections considered here.

We also tested three different variants of the language model (LM) on these collections. Table 3 shows the performance of these variants, where the first sub-column corresponds to a 0.5 jelinek mercer smoothing, the second sub-column to a $\mu = 1000$ Dirichlet Smoothing, and the third sub-column to a leave-one-out likelihood Dirichlet smoothing estimation [10]. As one can note, the best results obtained with LM models are either equal or slightly lower than the best results obtained with standard DFR and log-logistic models, on all collections.

Table 3. Mean Average Precision and Precision at 10 documents for language models with 3 differents settings: first sub-column corresponding to a 0.5 Jelinek-Mercer smoothing, second to a $\mu = 1000$ Dirichlet prior, and third to a Dirichlet prior estimated with the leave-one-out likelihood method

	TREC3-t	ROB-t	ROB-d	CLEF03	GIRT	TEL-BL
MAP	23.0 26.8 26.2	22.7 24.5 24.8	26.4 27.0 26.9	48.9 46.2 48.5	39.5 39.5 40.9	31.8 27.4 32.0
P10	43.8 53.4 54.4	39.3 43.9 44.2	44.3 45.7 45.5	35.0 32.5 34.0	65.6 67.3 68.7	47.8 44.4 49.0

It is interesting to note that model LGD outperforms model LG overall, which means that the estimation based on the document frequency (n_w) is better than the one based on the collection frequency (F_w). Language models are however unable to directly use document frequency information, since there is no direct way to convert this information into a probability distribution to be used as a collection model.

6 Conclusion

We have in this paper first introduced an analytical characterization of heuristic retrieval constraints and reviewed several DFR models wrt this characterization. This review showed that the first normalization principle of DFR is necessary to make the model compliant with retrieval constraints. We have then introduced a new model based on the log-logistic distribution to derive a simplified DFR model, and have shown that this simplified model contained, as a special case, the standard language model with Jelinek-Mercer smoothing. This relation is, to our knowledge, the first connection between the DFR and language modeling approaches to IR. Lastly, we have presented experimental results obtained on several standard collections which validate the simplification and the model we have introduced.

References

1. Airoldi, E.M., Cohen, W.W., Fienberg, S.E.: Bayesian methods for frequent terms in text: Models of contagion and the &delta;&sup2; statistic
2. Amati, G., Rijsbergen, C.J.V.: Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Trans. Inf. Syst. 20(4), 357–389 (2002)
3. Church, K.W., Gale, W.A.: Poisson mixtures. Natural Language Engineering 1, 163–190 (1995)
4. Clinchant, S., Gaussier, É.: The BNB distribution for text modeling. In: Macdonald, et al. (eds.) [7], pp. 150–161.
5. Airoldi, S.F.E.M., Cohen, W.W.: Statistical models for frequent terms in text. In: CMU-CLAD Technical Report - http://reports-archive.adm.cs.cmu.edu/cald2005.html (2004)
6. Fang, H., Tao, T., Zhai, C.: A formal study of information retrieval heuristics. In: SIGIR 2004: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (2004)
7. Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W.: ECIR 2008. LNCS, vol. 4956. Springer, Heidelberg (2008)
8. Na, S.-H., Kang, I.-S., Lee, J.-H.: Improving term frequency normalization for multi-topical documents and application to language modeling approaches. In: Macdonald, et al. (eds.) [7], pp. 382–393.
9. Singhal, A., Buckley, C., Mitra, M.: Pivoted document length normalization. In: SIGIR 1996: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 21–29. ACM, New York (1996)
10. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst. 22(2), 179–214 (2004)

A Analytical Characterization of IR Heuristics

We show here that the conditions we have presented in section 2 are closely related to the formal definitions of heuristic retrieval constraints given by Fang et al. [6]. The notation is the same as the one used before. In particular, a retrieval function, RSV, is defined by $RSV(d,q) = \sum_{w \in q \cap d} h(x_w^d, y_d, z_w, \theta)$. We first briefly recall the definitions of [6]:

TFC1: Let $q = w$ be a query with only one term w. Assume $y_{d1} = y_{d2}$. If $x_w^{d1} > x_w^{d2}$, then $RSV(d1,q) > RSV(d2,q)$.

TFC2: Let $q = w$ be a query with only one term w. Assume $y_{d1} = y_{d2} = y_{d3}$ and $x_w^{d1} > 0$. If $x_w^{d2} - x_w^{d1} = 1$ and $x_w^{d3} - x_w^{d2} = 1$, then $RSV(d2,q) - RSV(d1,q) > RSV(d3,q) - RSV(d2,q)$.

LNC1: Let q be a query and $d1$, $d2$ two documents. If for some word $w' \notin q$, $x_{w'}^{d2} = x_{w'}^{d1} + 1$ but for any query term w, $x_w^{d2} = x_w^{d1}$, then $RSV(d1,q) \geq RSV(d2,q)$.

LNC2: Let q be a query. $\forall k > 1$, if $d1$ and $d2$ are two documents such that $y_{d1} = k \times y_{d2}$ and for all terms w, $x_w^{d1} = k \times x_w^{d2}$, then $RSV(d1,q) \geq RSV(d2,q)$.

TF-LNC: Let $q = w$ be a query with only one term w. If $x_w^{d1} > x_w^{d2}$ and $y_{d1} = y_{d2} + x_w^{d1} - x_w^{d2}$, then $RSV(d1, q) > RSV(d2, q)$.

TDC: Let q be a query and $w1$, $w2$ be two query terms. Assume $y_{d1} = y_{d2}$, $x_{w1}^{d1} + x_{w2}^{d1} = x_{w2}^{d1} + x_{w2}^{d2}$. If *idf(w1)* $\geq$ *idf(w2)* and $x_{w1}^{d1} \geq x_{w1}^{d2}$, then $RSV(d1, q) \geq RSV(d2, q)$.

An interesting special case of TDC corresponds to the situation where $w1$ only occurs in $d1$ and $w2$ only in $d2$. With this setting, the constraint can be formulated as:

speTDC: Let q be a query and $w1$, $w2$ two query terms. Assume $y_{d1} = y_{d2}$, $x_{w1}^{d1} = x_{w2}^{d2}$, $x_{w1}^{d2} = x_{w2}^{d1} = 0$. If *idf(w1)* $\geq$ *idf(w2)*, then $RSV(d1, q) \geq RSV(d2, q)$.

The following property provides an analytical characterization of the above constraints (s.c. stands for *sufficient condition*).

Property 1. Let:

$$\begin{aligned}
&\forall(y, z, \theta),\ n \in \mathbb{N}^*,\ a_n = h(n, y, z, \theta)\\
&\forall(x, z, \theta),\ n \in \mathbb{N}^*,\ b_n = h(x, n, z, \theta)\\
&\forall(y, z, \theta),\ n \in \mathbb{N}^*,\ c_n = h(n+1, y, z, \theta) - h(n, y, z, \theta)
\end{aligned}$$

(i) TFC1 $\Longleftrightarrow a_n$ increasing. A s.c. is: $\forall(y, z, \theta),\ \frac{\partial h(x,y,z,\theta)}{\partial x} > 0$

(ii) TFC2 $\Longleftrightarrow c_n$ decreasing. A s.c. is: $\forall(y, z, \theta), \frac{\partial^2 h(x,y,z,\theta)}{\partial x^2} < 0$

(iii) LNC1 $\Longleftrightarrow b_n$ decreasing. A s.c. is: $\forall(x, z, \theta), \frac{\partial h(x,y,z,\theta)}{\partial y} < 0$

(iv) LNC2 $\Longleftrightarrow \forall(z, \theta),\ (m, n) \in \mathbb{N}^*,\ k > 1,\ h(km, kn, z, \theta) \geq h(m, n, z, \theta)$

(v) TF-LNC $\Longleftrightarrow \forall(z, \theta),\ (m, n, p) \in \mathbb{N}^*,\ h(m+p, n+p, z, \theta) > h(m, n, z, \theta)$

(vi) *speTDC* $\Longleftrightarrow \forall(x, y, \theta), \frac{\partial h(x,y,z,\theta)}{\partial z} < 0$

As mentioned before, conditions (i), (iii) and (vi) directly state that h should be increasing with the term frequency, and decreasing with the document length and the document/collection frequency. Note that condition (vi) only represents a necessary condition for the constraint TDC, as we have considered here a restricted form (speTDC) of this contraint. Condition (iv) and (v) directly regulate the interaction between the term frequency and the document length. Lastly, conditions (i) and (ii) state that h should be an increasing, concave function of the term frequency, the concavity ensuring that the increase in the retrieval score will be smaller for larger term frequencies.

Ordinal Regression Based Model for Personalized Information Retrieval

Mohamed Farah

Riadi-Gdl, Faculté des Sciences de Monastir, Tunisia
mohamed.farah@riadi.rnu.tn

Abstract. Retrieving relevant items as a response to a user query is the aim of each information retrieval system. But 'without an understanding of what relevance means to users, it is difficult to imagine how a system can retrieve relevant information for users' [1]. In this paper, we try to capture what relevance is for a particular user and model his profile implicitly considering his non declared preferences that are inferred from a ranking of a *reduced* set of retrieved documents that he produces. We propose an ordinal regression based model for interactive ranking which uses both the information given by this subjective ranking, as well as the multicriteria evaluation of these ranked documents, to adjust optimally the parameters of a ranking model. This model consists of a set of additive value functions which are built so as they are as compatible as possible with the subjective ranking. The preference information used in our model requires reasonable cognitive effort from the user.

Keywords: Ordinal Regression, UTA Method, Multiple Criteria Analysis, Interactive Information Retrieval Model, Aggregation, Personalization, Implicit User Profile, Relevance Feedback.

1 Introduction

According to [2], user's *Information Need* (IN) receives many transformations before being submitted to an Information Retrieval System (IRS). In fact, when a user is in a problematic situation, he needs information beyond his knowledge to solve his problem. He then perceives the IN and builds a mental representation of it, called the *Perceived Information Need* (PIN) which is an implicit representation in the user's mind. Next, the user expresses this PIN in a *Request* (RQ) which is a representation of the PIN in a human language, usually in natural language. Finally, he transforms RQ in the IRS language which is mainly boolean. This is the *Query* (Q) that he submits to the IRS.

These four levels are ordered in decreasing abstraction levels, i.e. if $\succ$ represents the binary relation whose semantics is *'more abstract than'*, then we have IN $\succ$ PIN $\succ$ RQ $\succ$ Q. Moving from one level to a less abstract one is necessarily accompanied with some *drift* from the original need that is caused by the *perception* of the IN, the *translation* of the PIN, or the *formulation* of the RQ.

Considering the information objects that IRSs try to retrieve, we can distinguish 3 different abstraction levels. In fact, each author, when publishing his

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 66–78, 2009.

document, has some knowledge or information in mind (K) that he wants to disseminate. The physical entity corresponding to the document is in fact a specific formulation of K, namely DR. Also, documents are not stored in the DR form in IRSs. They are rather indexed and stored differently, namely as document surrogates (DS), such as inverted indexes, titles, abstracts, keywords, snippets, anchor texts, etc. Therefore, we have K $\succ$ DR $\succ$ DS.

The aim of any IRS is to find relevant items meeting information seeker's need. Nevertheless, a fundamental question is: what kind of relevance do IRSs try to optimize ? and is it the relevance that users mean ?

Although research in information retrieval (IR) began during the 1950s, relevance still remains a crucial and much debated concern of IR, thus a major area of research in the field. One noticeable conclusion that can be drown from these studies is that there is no one single relevance but rather a system of relevances where each definition depends on factors such as i) the query level (IN, PIN, RQ, Q) [3], ii) the document level (K, DR, DS) [4], iii) the considered sources of evidence (the term frequency *tf*, the inverse document frequency *idf*, the document length, etc.) [5], and iv) the way all these sources of evidence are combined to produce the final ranking, i.e. the IR models (Boolean, Vector Space, Probabilistic, Language, etc. [6]).
Formally, let

- $N = \{IN, PIN, RQ, Q\}$ be the different user's information need levels,
- $D = \{K, DR, DS\}$ be the different author's information object levels,
- F be the set of factors or criteria that IRSs use to assess relevance, and
- M be the set of IR models that encompass the way the preceding factors are combined to compute a relevance score.

For each combination of these elements, there exists one definition for relevance. The set Rel of all the kinds of relevances is:

$$Rel = N \times D \times F \times M \tag{1}$$

In [7], relevance is a relation between some form of the information need N and the document D. This definition is coherent with the model of equation (1) when we suppose that giving some information need and a document entity, there exists an optimal combination of $F \times M$. In this vein, [7] proposed a typology of relevances that spans from a system oriented definition, namely a *system or algorithmic relevance* to a more personalized one, namely *cognitive relevance* or *pertinence*.

According to this framework, most IRSs aim to maximize an algorithmic relevance by retrieving document's surrogates (DS) matching the query (Q). Therefore, they basically differ w.r.t. the factors used to assess relevance as well as the way these factors are combined.

Relevance feedback (RF) is a wide area of research in IR that could be considered as an attempt to move from a systems-centered to a user-centered definition of relevance. Most of RF techniques iteratively modifies the description of the information need, i.e. the query, by adding or discarding terms or expressions. Thus, a new query is formulated by adding the selected terms for a second round retrieval. Through query expansion, some relevant documents missed in the initial round can then be hopefully retrieved to improve the overall system performance. Put differently, RF, in its current form, modifies the N component of the $N \times D \times F \times M$ model.

In this paper, we introduce a new way of dealing with relevance information by optimizing the M component of the $N \times D \times F \times M$ model. We propose a semi-supervised extension of the multicriteria model of [8] that incorporates relevance information. The resulting interactive model aims for a more personalized information retrieval and is based on ordinal regression methods. In this model, the learning phase is used for the tuning of the aggregation function combining document's performances on a set of relevance criteria. The novelty of our approach is that it is a way for modeling the user profile implicitly, i.e. considering non declared preferences that are inferred from a ranking of a reduced set of retrieved documents.

The paper is organized as follows. We first report related work in Section 2. The multicriteria model [8] is then briefly described in Section 3. Our interactive multicriteria model in presented in Section 4. Conclusions and avenues for future research are provided in the final section (Section 5).

2 Related Work

This paper is concerned with how to deal with relevance information. Therefore we report hereafter related research in the field of relevance feedback.

There are three families of relevance feedback techniques [9,10]: explicit feedback, implicit feedback, and blind or pseudo relevance feedback.

2.1 Explicit Feedback

It is a process in which the user conducts an initial query, then provides explicit relevance feedback. We distinguish two categories of explicit feedback depending on whether relevance judgments concern documents or terms.

In the first case, the user has to assess some retrieved documents as relevant or not. Afterwards, additional terms from these documents are used to compute a new query formulation for a new round retrieval. A second class of explicit feedback, called search suggestion or search assistance, consists of allowing users choose relevant terms that are automatically computed beforehand. These terms are afterwards used in order to compute a new query formulation for a new retrieval. It is used by several search engines which try to assist users refine their queries by providing related topics. In both cases, the process can also go through one or more iterations until the user is satisfied.

2.2 Implicit Feedback

Implicit or indirect feedback is inferred from user behavior, such as noting which documents are selected for viewing, the duration of time spent viewing a document, noting page browsing or scrolling actions, or using techniques such that eye-tracking. For instance, on the web, DirectHit ranks more highly documents which are viewed by users more often. Also, Clickstream data, i.e. data about what links a user clicks on, also provides indirect relevance feedback. Even the link structure of the collection, used in the PageRank algorithm, can be viewed as implicit feedback, but provided by page authors rather than readers.

2.3 Blind Feedback

Blind feedback or pseudo relevance feedback is obtained by assuming that the top k retrieved documents in the result set of cardinality n (usually $k \ll n$) are relevant. Therefore, blind feedback automates the manual part of explicit feedback and thus there is no need for user intervention. This technique mostly works, especially in the TREC ad hoc task. Nevertheless, as a full automatic process, it can cause query drift in a wrong direction, as unrelated terms are added to the query.

A variant of pseudo-relevance feedback, called ‘Real-Time Query Expansion’ (RTQE), consists of suggesting additional query terms while the user is typing his query, i.e. offering query expansion facility while the query is formulated. Similar techniques have already been implemented in commercial search engines, such as the Google Suggest.

Whatever is the query expansion technique, relevance information is used in order to either adjust the weights of terms in the original query, or to add or discard terms to/from the initial query. Relevance feedback was first implemented using the Rocchio algorithm [11].

3 The MultiCriteria Model for IR

Here, we briefly describe the MultiCriteria Model for IR (MCM-IR). Only relevant elements of the model are reported here. For a more comprehensive presentation of the model, we refer the interested reader to [8].

In the MCM-IR model, relevance is defined by a set of criteria and ranking is derived from pairwise comparisons of document performance vectors (*document profiles*) using decision rules identifying positive and negative reasons for judging whether or not a document should get a better ranking than another.

Let us mention that the MCM-IR model, as several retrieval models, aims to maximize an algorithmic relevance, and therefore does’t involve any kind of user preferences or needs.

The overall approach is split into four phases:

- The *modeling phase* consists in identifying various factors affecting relevance. These factors are used to develop a set of appropriate decision criteria which

model different aspects of relevance. Each criterion will give rise to a *partial preference relation* (binary relation) modeling the way two documents are compared, according to that criterion.
- The *filtering phase* aims at identifying a reduced set of *potentially high relevant documents.* To do so, we use a profile-based filter which selects documents that satisfy an *acceptance profile* defined by minimal requirements on the values of some or all criteria.
- The *aggregation phase* aggregates partial preference relations derived from pairwise comparisons of documents w.r.t. each criterion, into one or more *global preference relation(s).* A global preference relation indicates how two documents are compared w.r.t. all the considered criteria.
- The *exploitation phase* processes global preference relations resulting from the previous phase in order to derive the final ranking of documents.

3.1 Modeling Phase

A criterion is the basis for partial relevance judgments as to whether a document is more or less relevant than another w.r.t. a specific point of view. It is modeled by a real-valued non-decreasing function g defined on the set of documents which aims at comparing any pair of documents d_j and d_k, on a specific point of view, as follows:

$$g(d_j) \geq_i g(d_k) \Rightarrow d_j \text{ `is at least as relevant as' } d_k \text{w.r.t. criterion} g_i$$

Since, many formulations of each criterion are possible, we should not overemphasize the criterion scores of documents. Thus, it is often inadequate to consider that slight differences in evaluation should give rise to clear-cut distinctions.

Imprecision underlying criteria design is modeled using 2 discrimination thresholds [12]:

- An *indifference threshold* allows for two close-valued documents to be judged as equivalent although they do not have exactly the same score on the criterion. The indifference threshold basically draws the boundaries between an indifference and a preference situations.
- A *preference threshold* is introduced when we want or need to be more precise when describing a preference situation. Therefore, it establishes the boundaries between a situation of a strict preference and an hesitation between an indifference and a preference situations, namely a weak preference.

A criterion g_i, having indifference and preference thresholds q_i and p_i ($p_i \geq q_i \geq 0$) is called a *pseudo-criterion.* Comparing two documents d_j and d_k according to a pseudo-criterion g_i leads to the following partial preference relations:

$$\begin{aligned} d_j I_i d_k &\Leftrightarrow -q_i \leq g_i(d_j) - g_i(d_k) \leq q_i \\ d_j Q_i d_k &\Leftrightarrow q_i < g_i(d_j) - g_i(d_k) \leq p_i \\ d_j P_i d_k &\Leftrightarrow g_i(d_j) - g_i(d_k) > p_i \end{aligned}$$

where I_i, Q_i and P_i represent respectively *indifference*, *weak preference* and *strict preference relations* restricted to criterion g_i. These 3 relations could be grouped into an *outranking relation* $S_i = (I_i \cup Q_i \cup P_i)$ such that $d_j S_i d_k \Leftrightarrow g_i(d_j) - g_i(d_k) \geq -q_i$ which corresponds to the assertion d_j *'is as least as relevant as'* d_k w.r.t. the aspects covered by criterion g_i.

To model situations where a very low score of a document d_k w.r.t. d_j according to some criterion g_i cannot be compensated by good scores on one or several other criteria, we use a *veto threshold* v_i $(v_i \geq p_i)$ and define the following *veto relation* $V_i : d_j V_i d_k \Leftrightarrow g_i(d_j) - g_i(d_k) > v_i$. In this case, d_k cannot be considered as *'at least as relevant as'* d_j.

3.2 Ranking Procedure

In order to get a global relevance model on the set of documents, *outranking approaches* are used [13]. They are based on a *partial compensatory logic* and consist of two phases: an aggregation phase and an exploitation phase.

Aggregation Phase: Outranking approaches take as input the partial preference relations induced by the criteria family and aggregate them into one or more global preference relation(s) S. They are particularly relevant since they (i) permit considering imprecision in document evaluations, (ii) can handle criteria expressed on heterogeneous scales, (iii) use all the available information on document performances, and (iv) do not necessarily require inter-criteria information such as weights.

In order to accept the assertion $d_j S d_k$, stating that 'document d_j is at least as relevant as document d_k', the following conditions should be met:

- a *concordance* condition which ensures that a majority of criteria are concordant with $d_j S d_k$ (*majority principle*).
- a *discordance* condition which ensures that none of the discordant criteria strongly refutes $d_j S d_k$ (*respect of minorities principle*).

In the original paper, authors suppose that there is no information on the *relative importance of criteria*. Thus, to accept assertions like $d_j S d_k$, they use decision rules based on the criteria *supporting* (positive reasons) or *refuting* (negative reasons) this assertion. They use embedded outranking relations such as:

$$\begin{aligned} &d_j S^1 d_k \Leftrightarrow C(d_j S d_k) = F \\ &d_j S^2 d_k \Leftrightarrow c(d_j P d_k) \geq c(d_j P^- \cup Q^-\, d_k) \quad \text{and} \quad C(d_j V^- d_k) = \emptyset \end{aligned}$$

where

- $F = \{g_1, \ldots, g_p\}$ is a family of p criteria,
- P, Q, V and S are global preference relations,
- H^- is a relation such that $d_j H^- d_k \Longleftrightarrow d_k H d_j$,
- H_i is a partial preference relation, i.e. restricted to criterion g_i,

- $C(d_jHd_k) = \{g_i \in F : d_jH_id_k\}$ is the concordance coalition of criteria in favor of establishing d_jHd_k, and
- $c(d_jHd_k)$ is the number of items in $C(d_jHd_k)$

The S_1 relation between documents d_j and d_k holds if all the criteria are concordant with d_jSd_k. To accept $d_jS^2d_k$, there should be more criteria concordant with d_jPd_k than criteria supporting a strict or weak preference in favor of d_k. At the same time, no discordant criterion should strongly disagree with this assertion.

Exploitation Phase: Outranking relations resulting from the preceding phase are not necessarily transitive and do not lend themselves to immediate exploitation to get the final ranking. Therefore, complementary procedures, called exploitation procedures, are used in order to derive the final document ranking. In ibidem, authors propose a procedure which consists in iteratively partitioning the set of documents into *r ranked classes* $C_1, \ldots, C_r$ where each class C_h contains documents with the same score. Documents from class C_h are more relevant than documents from class C_{h+1}, $\forall h = 1, \ldots, (r-1)$.

4 Ordinal Regression Based Model for Personalized IR

Although the MCM-IR model is very useful for IR, it suffers from two major drawbacks. First, implementing the model is too time and space consuming, especially for building the outranking relations in the modeling and aggregation phases. This is nevertheless common for all pairwise approaches. To make the model tractable, authors [8] propose a filtering phase which consists of discarding documents w.r.t. their performances in the criteria manifold. Also, considering there is no information on the relative importance of criteria, authors suppose each criterion is neither prevailing, nor negligible. Thus, the response to a request is not user sensitive.

In this paper, we suppose that relevance is modeled by a set of relevance criteria as in the MCM-IR model. Nevertheless, while the MCM-IR model uses outranking approaches in the aggregation phase, we propose a completely different theoretical framework to deal with such aggregation: we use ordinal regression which aggregate document performances in a complete compensatory logic. Moreover, our model allows considering information about user preferences.

Our interactive model alternates a *dialog* or *interaction phase* where a user interacts with the system by ranking a reduced set of retrieved documents, and a *computation phase* where the potentially relevant items are re-ranked according to the user feedback as well as to document performances w.r.t. a set of relevance criteria. During this last phase, model parameters are tuned in order to best fit the user feedback. The process can go through one or more iterations of the sort.

From the theoretical point of view, similar approaches dealing with personalization do exist in the decision theory [14] as well as in combinatorial optimization literature [15]. These approaches basically suppose that the decision maker (the user in our context) is able to specify a typical decision alternative or solution (a typical relevant document in our context) which is often a non realistic or imaginary alternative, called the *reference point.* Real alternatives can therefore be ordered in decreasing order of the similarity to that reference point.

Approaches of this family iteratively compute the best alternative (choice problem) using either the L_1-norm (weighted sum average) [16] or the L_∞-norm (the Tchebytchev distance) [17].

In the IR context, users are more familiar with the *ranking philosophy*, i.e. they can easily rank a reduced set of documents from the most relevant one to the least relevant one. Moreover, it is rather difficult to imagine how a user can specify a *reference document* very specifically, i.e. a document with specific quantitative and qualitative performances in the criteria manifold. In fact, this needs the user to make unrealistic assertions like 'the best document should have $tf = 12$' where tf is the term frequency. Besides, it is also difficult to imagine the user finding a typical relevant document (holistic judgment) unless he reads all the retrieved documents R, which is obviously unrealistic.

Considering these findings, we propose an interactive ranking procedure, called the Ordinal Regression Based Model for Personalized IR (ORBM-PIR), which finds its roots in the UTA method [18,19] for ordinal regression and which uses both the information given by a subjective ranking on a set of documents given by a user, as well as the multicriteria evaluation of these documents, to adjust optimally or infer the parameters of the ranking model such that it is as consistent as possible with the subjective ranking.

Our method assumes that there exists a non-decreasing *marginal utility function* $u_i(g_i) = u_i$ corresponding to each criterion g_i as well as an *additive utility function* U [20] that encompasses the ranking model, i.e.

$$U(d_j) = \sum_{i=1}^{p} u_i(g_i(d_j)) \tag{2}$$

More formally, let $R = \{d_1, d_2, \ldots, d_n\}$ denote the set of retrieved documents matching a query which is evaluated on a family $F = \{g_1, g_2, \ldots, g_p\}$ of p criteria. F is supposed to satisfy consistency conditions [13], i.e. completeness (all relevant criteria are considered), monotonicity (increasing the evaluation of a document on some criterion leads to increasing its relevance to a query), and non-redundancy (no superfluous criteria are considered). Let $g(d_j) = [g_1(d_j), \ldots, g_p(d_j)]$ be the multicriteria evaluation vector of document d_j. We assume, without loss of generality, that the greater is $g_i(d_j)$, the better is document d_j on criterion g_i.

Let $g_{i*} = \min_j\{g_i(d_j)\}$ and $g_i^* = \max_j\{g_i(d_j)\}$ be respectively the worst and the best (finite) evaluations on g_i. $E_i = [g_{i*}, g_i^*]$ is the scale of criterion g_i, i.e. the range in which the values of criterion g_j are found. Consequently, the evaluation space is $E = \prod_{g_i \in F} E_i$ and $g(d_j) \in E$ is a profile in such space E.

User preferences are elicited in the form of a ranking that can be modeled using 2 global preference relations: an indifference relation I and a strict preference relation P. Therefore, the following generally holds for U:

$$d_j P d_k \Leftrightarrow U(d_j) > U(d_k) \Leftrightarrow \sum_{i=1}^{p} u_i(g_i(d_j)) > \sum_{i=1}^{p} u_i(g_i(d_k))$$
$$d_j I d_k \Leftrightarrow U(d_j) = U(d_k) \Leftrightarrow \sum_{i=1}^{p} u_i(g_i(d_j)) = \sum_{i=1}^{p} u_i(g_i(d_k))$$

The subjective ranking is therefore a *complete preorder* $S = (P, I)$ on a reduced subset $\widetilde{R} \subset R$ of documents with multicriteria evaluations on E. I and P are respectively the symmetric and asymmetric parts of this preorder that we call hereafter the *reference preorder*.

As in the UTA method, we consider that for each criterion g_i, the corresponding marginal utility function u_i is estimated by a piecewise linear function. Thus, the range E_i is divided into $\alpha_i \geq 1$ equal sub-intervals $[g_i^k, g_i^{k+1}], \forall k = 1, \ldots, (\alpha_i - 1)$ where α_i is a parameter. If E_i is discrete, α_i can be set to the number of grades of the interval E_i or a subset of these grades. Therefore, each end point g_i^k is given by the following formula:

$$g_i^k = g_{i*} + \frac{k-1}{\alpha_i - 1}(g_i^* - g_{i*})$$

Estimating the u_i functions is equivalent to estimating the variables $u_i(g_i^k) = u_i^k$. Therefore, the marginal utility of a document d_j w.r.t. criterion g_i, is approximated by a linear interpolation as shown in Figure 1. Thus, for $g_i(d_j) \in [g_i^k, g_i^{k+1}]$, we have:

$$u_i(g_i(d_j)) = u_i^k + \frac{g_i(d_j) - g_i^k}{g_i^{k+1} - g_i^k}(u_i^{(k+1)} - u_i^k) \tag{3}$$

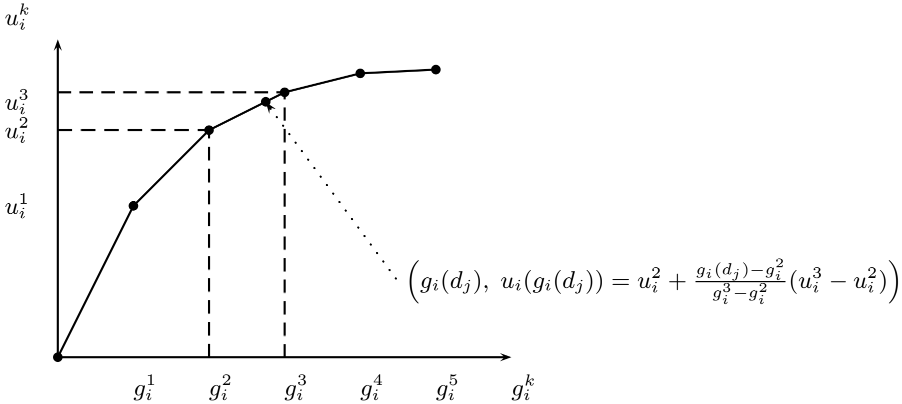


Fig. 1. Computation of the marginal utility of a document d_j w.r.t. criterion g_i

To find the variables u_i^k, we need to resolve the following linear program LP:

$$\begin{aligned}
&Min\ Z = \sum_{d_j \in \widetilde{R}} \sigma(d_j)\\
&s.t\\
&\qquad \sum_{i=1}^{p} (u_i(g_i(d_j)) - u_i(g_i(d_k))) + \sigma(d_j) - \sigma(d_k) \geq \delta; \forall (d_j, d_k) \in \widetilde{R}^2 : d_j P d_k\\
&\qquad \sum_{i=1}^{p} (u_i(g_i(d_j)) - u_i(g_i(d_k))) + \sigma(d_j) - \sigma(d_k) = 0; \forall (d_j, d_k) \in \widetilde{R}^2 : d_j I d_k\\
&\qquad u_i^{(k+1)} - u_i^k \geq s_i; \forall i = 1, \ldots, p; k = 1, \ldots, (\alpha_i - 1)\\
&\qquad \sum_{i=1}^{p} u_i(g_i^*) = 1\\
&\qquad u_i(g_{i*}) = 0; \forall i = 1, \ldots, p\\
&\qquad u_i^k \geq 0, \forall i = 1, \ldots, p; k = 1, \ldots, (\alpha_i - 1)\\
&\qquad \sigma(d_j) \geq 0; \forall d_j \in \widetilde{R}
\end{aligned}$$

In the preceding linear program LP, the first two family of constraints model the reference preorder S. Using transformations of equation (3), they only involve the principle variables u_i^k. The third family of constraints are set since u_i are supposed to be non-decreasing marginal utility functions. The 4th constraint as well as the 5th family of constraints are set for normalization purposes: documents scores will range from 0 to 1. The last two family of constraints specify that the principle variables u_i^k as well as the auxiliary variables $\sigma(d_j)$ are non-negative. Moreover, auxiliary variables $\sigma(d_j)$ model errors, δ is an arbitrary small positive value parameter so as to significantly discriminate two successive equivalence classes of $\widetilde{R}$, and s_i is an indifference threshold parameter defined on criteria g_i to model imprecision.

The linear program LP can be resolved using the Simplex algorithm. Besides, the structure of the preceding LP is such that it is more useful to solve the dual in order to save time and memory.

If the optimal solution is $Z^* = 0$, then there exists a least one utility function U compatible with the reference preorder S. When the optimal value $Z^* > 0$, then there is no value function U compatible with the reference preorder S. In such circumstances, we can pursue one of the following strategies:

- increase the number α_i of breakpoints g_i^k for one or several marginal utility functions u_i,
- ask the user to revise the reference preorder on $\widetilde{R}$, or
- search over the relaxed domain $Z \leq Z^* + \varepsilon$ an additive value function U giving a preorder $\widehat{S}$ on $\widetilde{R}$ which is sufficiently close to the reference preorder S, in the sense of Kendall-tau distance or Spearman-footrule distance. Branch and bound methods could be used here.

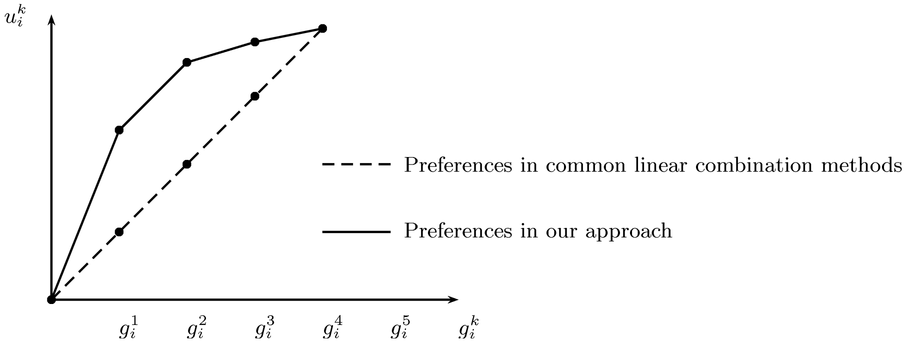


Fig. 2. Encoding of performances

The resulting solution is therefore used to compute the score of all the documents of R using formula of equations (2) and (3), and rank them accordingly. This guaranties that the resulting ranking is coherent with user preferences, thus personalized.

Before closing this section, we briefly report the main distinctive features of our ORBM-PIR model compared to the MCM-IR model of section 3 as well as common linear combination methods.

To rank documents, the MCM-IR model relies on the principle of pairwise comparison of alternatives, whereas our model is based on the idea of assigning a global score to each document, as in the Muti-Attribute Utility Theory (MAUT). Moreover, in the MCM-IR model, the criteria aggregation model is known a priori (the outranking aggregation approach) while the global preferences are unknown. On the contrary, our ORBM-PIR model involves the inference of the aggregation model which is unknown a priori, from global preferences that are elicited from the user.

Our model differs from common linear combination methods w.r.t. the following features. First, our model incorporates a semi-supervised learning phase which allows personalization. Moreover, linear combination methods consider that performances on each criterion g_i increases linearly all along the range E_i. This cannot model preferences corresponding to the following situations: suppose that a user prefers documents with more query terms occurrences up to a specific threshold k, after which, additional occurrences are meaningless. In our model, such configurations are possible as shown in Figure 2.

5 Conclusion

In this paper, we proposed an ordinal regression based model for personalized IR that allows inferring analytical aggregation models to be used to rank documents, based on implicit user preferences given by reference preorders on a reference set $\widetilde{R}$. Such aggregation model is produced after the user interacts with the IRS by picking few documents and ranking them w.r.t. his own perception of relevance.

The novelty of the approach is that it allows modeling the user profile by additive utility functions starting from a ranking of a reduced set of retrieved documents.

It is worthwhile noting that even in situations where users are unlikely to provide explicit feedback and assess the relevance of some returned documents, we can use implicit feedback techniques to deduce a personalized ranking. A straightforward strategy would be to rank documents in decreasing order of the duration of time spent viewing them.

Experimentation of the proposed model is ongoing research and will be reported in future publications.

Future research aims to consider situations when we have more information than the ranking itself. For example, when we have information related to the intensity of preferences, i.e. information expressed by assertions of the type 'd_j is preferred to d_k at least as much as d_l is preferred to d_m' whether expressed on particular criterion or in a holistic fashion. Also, it is interesting to device procedures that handle preference information with gradual credibility.

References

1. Schamber, L., Eisenberg, M., Nilan, M.: A re-examination of relevance: Toward a dynamic, situational definition. IPM 26(6), 755–776 (1990)
2. Taylor, R.S.: Question-negotiation and information seeking in libraries. College and Research Libraries 29, 178–194 (1968)
3. Belkin, N.J., Kantor, P., Fox, E.A., Shaw, J.A.: Combining evidence of multiple query representations for information retrieval. IPM 31(3), 431–448 (1995)
4. Katzer, J., McGill, M., Tessier, J., Frakes, W., DasGupta, P.: A study of the overlap among document representations. Information Technology: Research and Development 1(4), 261–274 (1982)
5. Lee, J.H.: Combining multiple evidence from different properties of weighting schemes. In: SIGIR 1995, pp. 180–188 (1995)
6. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval. ACM Press, New York (1999)
7. Saracevic, T.: Relevance reconsidered 1996. In: Information Science: Integration in Perspective, Proceedings of the CoLIS-2 conference. Royal School of Library and Information Science, Copenhagen, Denmark, pp. 201–218 (1996)
8. Farah, M., Vanderpooten, D.: An outranking approach for information retrieval. Information Retrieval 11(4), 315–334 (2008)
9. Ruthven, I., Lalmas, M.: A survey on the use of relevance feedback for information access systems. Knowl. Eng. Rev. 18(2), 95–145 (2003)
10. Manning, C.D., Raghavan, P., Schtze, H.: Relevance feedback and query expansion. In: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
11. Rocchio, J.: Relevance feedback in information retrieval, pp. 313–323. Prentice-Hall, Englewood Cliffs (1971)
12. Roy, B.: Main sources of inaccurate determination, uncertainty and imprecision. Mathematical and Computer Modelling 12(10-11), 1245–1254 (1989)
13. Roy, B.: The outranking approach and the foundations of ELECTRE methods. Theory and Decision 31, 49–73 (1991)

14. Vincke, P.: Multicriteria Decision-Aid. John Wiley and Sons, Chichester (1992)
15. Sawaragi, Y., Nakayama, H., Tanino, T.: Theory of Multiobjective Optimization, Orlando edn. Academic Press, London (1985)
16. Zionts, S., Wallenius, J.: An interactive programming method for solving the multiple criteria problem. Manage. Sci. 22(6), 652–663 (1976)
17. Benayoun, R., Laritchev, O., De Mongolfier, J., Tegny, J.: Linear programming with multiple objective functions: Step method (stem). Math. Program. 1(3), 366–375 (1971)
18. Jacquet-Lagrèze, E., Siskos, Y.: Assessing a set of additive utility functions for multicriteria decision making: the UTA method. European Journal of Operational Research 10, 151–164 (1982)
19. Jacquet-Lagrèze, E., Meziani, R., Slowinski, R.: Molp with an interactive assessment of a piecewise utility function. Eur. J. Oper. Res 31(3), 350–357 (1987)
20. Keeney, R., Raiffa, H.: Decisions with multiple objectives: Preferences and value tradeoffs. J. Wiley, New York (1976)

Navigating in the Dark: Modeling Uncertainty in Ad Hoc Retrieval Using Multiple Relevance Models

Natali Soskin, Oren Kurland, and Carmel Domshlak

Faculty of Industrial Engineering and Management, Technion, Haifa 32000, Israel
natalis@tx.technion.ac.il,{kurland,dcarmel}@ie.technion.ac.il

Abstract. We develop a novel probabilistic approach to ad hoc retrieval that explicitly addresses the uncertainty about the information need underlying a given query. In doing so, we account for the special role of the corpus in the retrieval process. The derived retrieval method integrates multiple *relevance models* by using estimates of their *faithfulness* to the presumed information need. Empirical evaluation demonstrates the performance merits of the proposed approach.

Keywords: relevance models, ad hoc retrieval, faithfulness measures.

1 Introduction

The ad hoc retrieval task is to find documents relevant to an information need expressed by a query. However, it is often a hard challenge to infer what the underlying information need is, especially in the case of ambiguous queries.

We present a novel probabilistic framework to ad hoc retrieval that *explicitly* addresses the uncertainty about the information need expressed by a query. In doing so we account for two major factors that affect uncertainty, namely (1) the fact that the same query can be used to represent different information needs, and (2) the "nature" of the corpus upon which the search is performed. A case in point for the latter, a query for the car Jaguar used over the Web should better include the term "car", yet this term has no discriminative power in a portal dedicated to cars. The retrieval model that we derive integrates *multiple* relevance models [1,2], e.g., statistical language models that are presumed to generate terms in relevant documents. These relevance models potentially correspond to information needs that may underlie the query.

Our framework can be instantiated in various ways to yield specific retrieval algorithms, varying, for example, in the set of relevance models considered and in the *faithfulness* we attribute to each of them with respect to an information need presumably represented by the query.

To exemplify the practical potential of our framework, we take a pseudo feedback approach, and construct multiple relevance models based on documents sampled from an initially retrieved list. We then propose several faithfulness measures. Empirical evaluation demonstrates the performance merits of our methods with respect to using a single relevance model.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 79–91, 2009.

2 Retrieval Framework

Taking a probabilistic approach to the task of ad hoc retrieval, our basic goal is to estimate the probability $p(d|q)$ that a given document d is relevant to a given query q. Since the relevance of a document should in fact be determined with respect to the information need I_q represented by q rather than with respect to q itself, it is important to reason about that information need within the process of estimating $p(d|q)$. The latter task is obviously challenging, because q can potentially represent different information needs, and, because in the ad hoc setting we usually do not have any information about I_q other than q.

Hence, while we assume that q communicates an (arbitrary complex, yet) *single* information need of the user, we should still strive to model and reason about our uncertainty on what that information need actually is.

Having this agenda in mind, we first consider the *generative assumption for relevance* [1,2] that states:

Assumption 1 (generative assumption). *Given information need I, there exists a relevance model R that generates the content in queries representing I and in documents relevant to I.*

The probabilistic graphical model representing this assumption is depicted in Fig. 1a. Henceforth, bold-faced and regular letters correspond to random variables and values of these random variables, respectively. The focus of prior work with respect to Assumption 1 was on estimating (some form of) a relevance model R by treating q as an observed sample from it; the documents are then ranked using (an estimate of) $p(d|R)$ [1]. However, as we argue next, the *estimation* of R touches only a part of the overall picture of the retrieval process.

We observe two "operational" aspects by which Assumption 1, as well as its practical realization described above, can be enhanced. The first aspect concerns the uncertainty about the information need underlying q. Prior work has coupled the information need with the relevance model, and addressed the uncertainty as an implicit part of estimating R. Thus, while the graphical model induces

$$p(d|q) = \sum_{I,R} p(d|R)p(R|I,q)p(I|q) \ ,$$

in practice, a single relevance model $R^* = \arg\max_R p(R|q)$ was *selected* by the virtue of choosing a specific estimation procedure for R, and then $p(d|R^*)$ served for $p(d|q)$. Note that doing so can conceptually be viewed as *replacing the true evidence q on $\mathbf{q}$ with a* de facto *evidence R^* on* $\mathbf{R}$, obtained by assuming, for example, (i) independence of $\mathbf{R}$ from $\mathbf{I}$ given q, and (ii) $p(\mathbf{I}|q)$ is a uniform distribution over those I that can represent q. In other words, in practice, the treatment of the uncertainty about the underlying information need is (implicitly) embodied in the specific choice of estimation technique for R^* rather than being reasoned about in the overall probabilistic model that $\mathbf{R}$ is a part of.

The second aspect is that of the context. Even if we were to know "exactly" what the information need is, the nature of the searched corpus C should have a

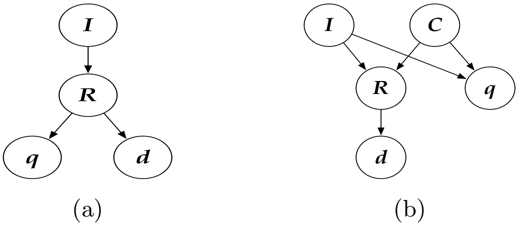


Fig. 1. Graphical-model representation of the (a) original, and (b) revised assumptions

significant impact on the way R is defined. For example, there might be language-specific issues (e.g., the terms "astronaut" and "cosmonaut" might need to be attributed with different levels of importance over English and Russian corpora, respectively). Moreover, a relevance model that can effectively discriminate (using some specific retrieval model) relevant documents from non-relevant ones over one corpus, might not be able to do so over a different corpus (e.g., recall the "Jaguar car" example from Sect. 1).

Given these two observations, we revise Assumption 1 as follows:

Assumption 2. *Given information need I and corpus C, there exists a relevance model R that generates documents relevant to I. Likewise, I and C determine the likelihood of a query q being selected to represent I in the context of C.*

Figure 1b depicts the corresponding graphical model. The major addition is modeling the dependence of $\mathbf{R}$ on the corpus $\mathbf{C}$. Note that the notion of corpus should be interpreted here not as a specific collection of documents, but rather as a *corpus characterization* (like "all documents on the Web" or "a collection of professional articles on various aspects of cardiology", etc.) Consequently, to represent the need for information about, for instance, "Jaguar cars", a relevance model R defined over the Web should better assign high importance to both the terms "Jaguar" and "car", while for R defined over a portal of cars, the term "car" should be assigned with low importance, if at all[1].

The second difference from Assumption 1 is that q is no longer assumed to be generated by R, but rather selected by a user to communicate her information need I in the context of C. For example, the user interested in a Jaguar car is less likely to use the single-term query "Jaguar" when searching over the Web, than when searching over a cars' portal. The reverse also holds, that is, the query "Jaguar" used over the Web should be considered by the search system to reflect a need for information with regard to both the car and the cat (potentially to varying degrees of importance) in lack of additional signal about the "true" need.

In the probabilistic model induced by Assumption 2, $\mathbf{R}$ still probabilistically depends on $\mathbf{q}$, but now via $\mathbf{I}$ and $\mathbf{C}$. That is, the observed query q reflects the latent information need I over C, while I and C determine the relevance model

[1] In practice, this could be done either implicitly (due to smoothing of document language models [3]), or explicitly [4].

R. We now turn to estimating $p(d|q)$ using Assumption 2. Specifically, we show that the indirect coupling between $\mathbf{R}$ and $\mathbf{q}$ via $\mathbf{I}$ and $\mathbf{C}$ allows for a direct reasoning about our uncertainty on the information need underlying q.

It is a fact that

$$p(d|q) = \sum_{C,I,R} p(d|R, I, C, q)p(I, C, R|q) \ ,$$

where the summation is over the universes of all corpora, information needs, and relevance models. Assumption 2 implies that the relevance of d to I can be determined based on R, and that R is uniquely determined given I and C. Thus,

$$p(d|q) = \frac{1}{p(q)} \sum_{C,I,R} p(d|R)p(R|I, C)p(I|q, C)p(q|C)p(C) \ ; \tag{1}$$

this equation calls for a closer examination.

The first component of the summation term, $p(d|R)$, provides the common ground between Assumptions 1 and 2 — it is the probability that d is generated from R, i.e., that d is relevant to the information need represented by R. The other components involve the corpus C and are therefore specific to Assumption 2. First, $p(R|I, C)$ is either 0 or 1, depending on whether R is the one corresponding to the given I and C. Next, $p(I|q, C)$ is the probability that I is the information need communicated by q in the context of C. This component has an interesting property that the *entropy* of $p(\mathbf{I}|q, C)$ is essentially the *"query difficulty"* [5]. That is, the closer the distribution $p(\mathbf{I}|q, C)$ to uniform (i.e., higher entropy), the more difficult it is to infer the information need underlying q, and consequently, the harder it is to distinguish between relevant and non-relevant documents. Indeed, some estimates for query difficulty (a.k.a. *query performance*) rely on the connection between q and C [5]. Finally, the probability $p(q|C)$ could be viewed as referring to the potential of having information in C that corresponds on a surface-level to q. For example, if q is written in a different language than that used in C, then this correspondence will be low. As a result, while the appropriate relevance model R for C will be described in terms of the language used in C (e.g., a cross-lingual relevance model [6]), the probability of relevance, $p(d|q)$, will be lower than that for a query that uses the same language used in the corpus.

Deriving estimates for some of the probabilities in Eq. 1, specifically, $p(d|\mathbf{R})$ and $p(\mathbf{I}|q, \mathbf{C})$, is obviously a hard task. Therefore, we make the following pragmatic assumptions and estimation choices. First, we use the standard relevance *language model* approach [1] for the estimate $\hat{p}(d|R)$ — i.e., we use the probability that terms in d are generated by a statistical language model representing R. Second, since not query difficulty but uncertainty about the information need is of our focus in this work, for $p(\mathbf{I}|q, C)$ we adopt a very simple estimate $\hat{p}(\mathbf{I}|q, C)$ corresponding to a uniform distribution *only* over information needs that *can* be represented by q over C. Next, we use q as a proxy for those I it can represent as it is the only piece of information we have with respect to the underlying information need in lack of an informative prior over information needs and/or additional user (relevance) feedback. Finally, we focus on the single-corpus search task (i.e., assume a single corpus C), and leave the multiple-corpora search task for future work, so as to arrive to the following rank equivalence:

$$\hat{p}(d|q) \overset{rank}{=} \sum_{R} \hat{p}(d|R)\hat{p}(R|q,C) \ . \tag{2}$$

It is important to note that while $p(R|I,C)$ is either 1 or 0, this is not the case for $\hat{p}(R|q,C)$ that results from using q as a proxy for I, as q can represent different information needs[2]. In what follows we treat $\hat{p}(R|q,C)$ as the probability that R corresponds to *some* information need I that is represented by q, where q is used to search information relevant to I over C.

2.1 Application

There are numerous ways of instantiating the ranking method presented in Eq. 2. However, to study the potential practical merits of the method, we need to make several implementation decisions. In what follows we present a possible set of such decisions, which constitutes only one example for how to use Eq. 2 to derive specific retrieval algorithms.

Our first task is to specify the set of relevance models to be utilized. There are various approaches for constructing relevance models (e.g, using documents [1], passages [7], document clusters [8], etc.). Here, we focus on utilizing documents for relevance-model construction.

Let $p(w|x)$ denote the probability assigned to term w by a (smoothed) unigram language model induced from text x. We use $\mathcal{D}_{init}^{[m]}$ ($\mathcal{D}_{init}$ in short) to denote the list of m documents d in the corpus C that yield the highest *query likelihood* $p(q|d) \overset{def}{=} \prod_{q_i} p(q_i|d)$; $\{q_i\}$ is the set of query terms. We define relevance model number 3 (RM3) [9] using the documents in $\mathcal{D}_{init}$:[3]

$$p(w|R) \overset{def}{=} \lambda p^{MLE}(w|q) + (1-\lambda) \sum_{d \in \mathcal{D}_{init}} p(w|d) \frac{\prod_{q_i} p(q_i|d)}{\sum_{d' \in \mathcal{D}_{init}} \prod_{q_i} p(q_i|d')} \ ; \tag{3}$$

$p^{MLE}(w|q)$ is the maximum likelihood estimate of w with respect to q; for $p(\cdot|d)$ we use a smoothed language model of d (further details in Sect. 4); λ is a free parameter.

[2] Since (i) R can represent different I's, and (ii) the estimate $\hat{p}(\mathbf{I}|q,C)$ is uniform over *only* those information needs that can be represented by q over C, $\hat{p}(\mathbf{R}|q,C)$ cannot be assumed to be uniformly distributed over the *entire* universe of relevance models.

[3] While RM3 assumes that q is generated from R this is not the case in Fig. 1b. We hasten to point out that using RM3 as an estimate for R here is only intended for performance-evaluation purposes, that is, to enable comparison of our paradigm that uses multiple relevance models with a state-of-the-art method that uses a single model. For full consistency with the graphical model, one could estimate R, for example, using a pseudo-feedback approach that only treats q as evidence for $\mathbf{I}$ (e.g., the state-of-the-art model-based feedback method [10]).

The list $\mathcal{D}_{init}$ often also contains non-relevant documents that may cause *query drift* [11] — i.e., shift between the information need underlying the query and that represented by the relevance model. Thus, as an alternative to using a single relevance model defined over $\mathcal{D}_{init}$, we define several relevance models that are constructed from documents sampled from $\mathcal{D}_{init}$. Specifically, we sample m sets of k documents (in Sect. 4 we compare random sampling [12] with cluster-based sampling [13]), and define over each set S a relevance model R_S using Eq. 3. Hopefully, some of the sampled sets will be composed of mainly relevant documents, or more generally, will faithfully reflect a "true" underlying information need. Naturally, the challenge, which we address below, is to quantify this faithfulness.

Our second order of business with respect to instantiating Eq. 2 is to estimate the probability of relevance model R generating the terms in document d, $\hat{p}(d|R)$. Some previous work [14] showed that in terms of retrieval effectiveness, using the cross-entropy between R and a language model induced from d is superior to estimating the probability that terms in d are generated from R. Thus, we use the complete-probability principle, and write:

$$\hat{p}(d|R) = \frac{\hat{p}(R|d)\hat{p}(d)}{\sum_{d'} \hat{p}(R|d')\hat{p}(d')} \quad . \tag{4}$$

We assume a uniform prior distribution for documents, $\hat{p}(\mathbf{d})$, and use a cross-entropy-based measure: $\exp(-CE\,(p(\cdot|R)\ ||\ p(\cdot|d)))$=$\exp(\sum_w p(w|R) \log p(w|d))$ for the estimate $\hat{p}(R|d)$. We note that while this measure does not constitute a probability distribution, the resultant estimate $\hat{p}(\mathbf{d}|R)$ in Eq. 4 does.

2.2 "Faithfulness" of Relevance Models

The last and most important task towards instantiating Eq. 2 is devising the estimate $\hat{p}(\mathbf{R}|q,C)$ of the probability that a relevance model R represents an information need underlying q with respect to C.

The estimate for relevance model R_S is $\hat{p}(R_S|q,C) \overset{def}{=} \frac{F(R_S;q,C)}{\sum_{S'} F(R_{S'};q,C)}$, where $F(R_S;q,C)$ is a real-valued function quantifying the extent to which R_S presumably represents, or in other words, is faithful to, an information need underlying q with respect to C. The first faithfulness measure that we consider is the **uniform** distribution that represents the belief that all constructed relevance models in $\{R_S\}$ are faithful to the same extent:[4]

$$F_{uniform}(R_S;q,C) \overset{def}{=} 1 \quad .$$

Next, the **constdoc** method estimates the faithfulness of R_S by the presumed percentage of relevant documents in S. Naturally, the more similar the constituent documents of S to the query are, the higher the estimate of this percentage should

[4] Note that there could be, and probably are, models in the universe of relevance models that are not in $\{R_S\}$ and that can faithfully represent the information need.

be. Following work on estimating the number of relevant documents in document clusters [15], we set:

$$F_{constdoc}(R_S; q, C) \overset{def}{=} \sqrt[|S|]{\prod_{d \in S} sim(q, d)} \ ,$$

where $sim(q, d) \overset{def}{=} \exp(-CE\left(p^{MLE}(\cdot|q) \,||\, p(\cdot|d)\right)) = \sqrt[|q|]{p(q|d)}$ is d's normalized query likelihood [16]; CE is the cross-entropy and $|q|$ is q's length.[5]

Both faithfulness functions just described consider the corpus only indirectly. We therefore study the **clarity** method [5], which is based on the KL divergence between R_s and the corpus model:

$$\begin{aligned} F_{clarity}(R_S; q, C) &\overset{def}{=} \exp(KL\left(p(\cdot|R_S) \,||\, p(\cdot|C)\right)) \\ &= \exp(\sum_w p(w|R_S) \log \frac{p(w|R_S)}{p^{MLE}(w|C)}) \ ; \end{aligned}$$

$p^{MLE}(w|C)$ is a maximum likelihood estimate of w with respect to C. The idea is that relevance models that are distant from the corpus model are "focused", and hence, are better candidates for representing a "coherent" information need [5]. Indeed, the value assigned by the clarity measure was shown to be somewhat correlated with the retrieval performance of the relevance model at hand [17].

The clarity measure does not consider (directly) the query for faithfulness estimation. The **drift** approach, in contrast, takes the query into account by measuring the divergence between the ranking induced by using R_S and that induced by using q. The idea is that the more distant the rankings are, the less faithful R_S is to the information need represented by q — i.e., the more chances there are for query drift [5,17]. The drift approach was shown to be effective for selecting a *single* relevance model from a set of candidates [17]. Formally, let L_q and L_{R_S} be the lists of 100 documents retrieved by using the original query q, and relevance model R_S, respectively. Let $p(w|L) \overset{def}{=} \beta \sum_{d_i \in L} p^{MLE}(w|d_i) + (1-\beta)p^{MLE}(w|C)$ be the language model induced from the document-list L; we set $\beta = 0.8$ [17]. The drift faithfulness measure is then:

$$\begin{aligned} F_{drift}(R_S; q, C) &\overset{def}{=} \exp(-CE\left(p(\cdot|L_q) \,||\, p(\cdot|L_{R_S})\right)) \\ &= \exp(\sum_w p(w|L_q) \log p(w|L_{R_S})) \ . \end{aligned} \tag{5}$$

3 Related Work

Some previous work is conceptually similar to ours in that it addresses the uncertainty with respect to the information need by using multiple (manually-created) query representations [18,19,20]. However, no probabilistic framework

[5] Using the arithmetic mean of the document-query similarity values yields performance inferior to that of using the geometric mean.

was presented, and the "faithfulness" of a query representation to the underlying information need was not modeled.

Recent work [17] selects a single relevance model, using the clarity and drift measures, from a set of models constructed from the initial list $\mathcal{D}_{init}$. In Sect. 4 we demonstrate the merits of our approach with respect to this paradigm.

There are various methods — including document re-sampling as we use here [12,13] — for improving the retrieval effectiveness of relevance models (e.g.,[7,8,4,13]), and of query-expansion models that could be viewed as relevance models (e.g., [10,12]). These methods produce a single relevance model used for ranking, in contrast to our approach that uses multiple relevance models for ranking. However, our approach can potentially use these methods as it is not committed to a specific paradigm of relevance-model estimation.

4 Evaluation

4.1 Experimental Setup

We conducted experiments on four TREC data sets: (i) AP (disks 1-3, topics 51-150), (ii) SJMN (disk 3, topics 51-150), (iii) WSJ (disks 1-2, topics 151-200), and (iv) ROBUST (disks 4,5 (-CR), topics: 301-450, 601-700). Topics titles served as queries. We applied tokenization, Porter stemming, and stopword removal (using the INQUERY list) via the Lemur toolkit[6], which was also used for retrieval.

Unless otherwise specified, we use Dirichlet-smoothed unigram document language models with the smoothing parameter value set to 1000 [3]. The query-likelihood model [21], **QL**, in which document d is scored by $p(q|d) \overset{def}{=} \prod_{q_i} p(q_i|d)$ — i.e., the surface-level similarity between q and d — serves as a reference comparison to the algorithms we explore.

We use MAP (at cutoff 1000) and the precision of the top 5 documents (p@5) for performance evaluation. Statistically-significant differences of performance are determined using the two-tailed Wilcoxon test at the 95% confidence level.

An additional reference comparison to our methods is **RelModel** — a relevance model constructed from all documents in the initial list $\mathcal{D}_{init}$, which was retrieved using the query likelihood (QL) method; m, the number of documents in $\mathcal{D}_{init}$, is set to 50, as is the case for all other methods. The other free parameters that RelModel incorporates are set to values so as to optimize MAP. Specifically, λ, which controls the reliance on the original query model, is set to values in $\{0, 0.2, \ldots, 1\}$; the Jelinek-Mercer smoothing parameter of the language models of documents in $\mathcal{D}_{init}$ is chosen from $\{0.1, 0.2, \ldots, 1\}$; and, the number of terms used by the relevance model is set to values in $\{5, 10, 25, 50, 75, 100, 500\}$. The documents in the corpus are ranked in RelModel by the cross-entropy between the relevance model and their Dirichlet-smoothed language models.

To create the document sets $\{S\}$ upon which the multiple relevance models are constructed, we employed either nearest-neighbor-based clustering over $\mathcal{D}_{init}$ (with the KL-divergence as a similarity measure) in which each document served

[6] www.lemurproject.org

Table 1. Performance numbers. Best result in a column is boldfaced. Statistically significant differences with QL and RelModel are marked with 'l' and 'r', respectively

	AP		SJMN		WSJ		ROBUST	
	MAP	p@5	MAP	p@5	MAP	p@5	MAP	p@5
QL	22.4	45.1	19.3	33.2	32.7	55.6	25.5	**48.2**
RelModel	28.9^l	50.7^l	24.1^l	38.4^l	38.7^l	**59.2**	27.6^l	46.9
uniform(rand)	28.6^l	50.7^l	23.4^l_r	38.4^l	38.3^l	58.8	27.1^l_r	47.7
uniform(clust)	29.3^l	52.7^l	24.5^l	39.8^l	39.2^l	57.2	26.9^l	46.0
constdoc(rand)	28.6^l	50.7^l	23.5^l	38.6^l	38.3^l	58.8	27.2^l_r	47.7
constdoc(clust)	$\mathbf{29.5}^l$	52.7^l	24.5^l	39.0^l	39.4^l	56.8	$\mathbf{28.4}^l$	47.9
clarity(rand)	28.7^l	50.5^l	23.6^l	39.0^l	38.3^l	58.0	27.0^l_r	47.7
clarity(clust)	29.3^l	52.9^l	24.6^l	39.6^l	$\mathbf{39.6}^l$	58.0	27.1^l_r	44.7
drift(rand)	28.6^l	50.9^l	23.5^l	38.8^l	38.2^l	58.4	27.1^l_r	**48.2**
drift(clust)	29.3^l	$\mathbf{53.1}^l$	$\mathbf{24.9}^l_r$	$\mathbf{40.8}^l_r$	39.2^l	58.0	27.7^l	47.9

as a basis for a cluster, or random selection from $\mathcal{D}_{init}$. In each case, 50 sets of k documents are used. Experiments with $k \in \{5, 10, 20\}$ under cluster-based selection showed that clusters of 5 and 10 documents yield relatively the same performance, while those of 20 documents yield inferior performance; hence, we set $k = 10$ in *all* tested models.

For computational reasons, we use each relevance model R to retrieve 1000 documents. Then, the lists retrieved by the multiple relevance models are *fused* using Eq. 2. Note that doing so simply amounts to setting $\hat{p}(R|d) = 0$ for all but 1000 documents d that yield the highest $\hat{p}(R|d)$. The other free parameters used to construct the multiple relevance models (λ, Jelinek-Mercer smoothing parameter, and number of terms) are set to the values chosen for RelModel, with which we compare our models as mentioned above. Hence, our multiple-relevance-models implementations are considerably *underoptimized* with respect to RelModel, as our goal is to focus on the underlying principles of our approach rather than engage in excessive tuning of parameters' values.

We use F(M) to denote a multiple-relevance-models implementation that uses the faithfulness measure F∈ {uniform, constdoc, clarity, drift} and the selection method M — either cluster-based (clust) or random-based (rand).

4.2 Results

Table 1 presents the performance numbers of our methods. Our first observation is that in most reference comparisons (corpus × evaluation measure) cluster-based selection of documents yields better performance than random-based selection. (Compare F(clust) with F(rand) rows.) This finding attests to the merit in constructing relevance models based on sets of similar documents that are potentially topically related. In addition, we note that both random-based and cluster-based implementations yield performance that is better (often to a statistically significant degree) than that of QL — the language model baseline.

The drift(clust) implementation yields, in general, the most effective performance among the implementations we consider. Thus, the divergence between the ranking induced by a relevance model and that induced by using the original

Table 2. Comparison with cluster-based document re-sampling (CBRSD) [13] for relevance-model construction. '>QL': percentage of queries for which the performance transcends that of QL. Boldface: best result in a column. Statistically significant differences with QL, RelModel, and CBRSD are marked with 'l', 'r', and 'c', respectively.

	AP				SJMN			
	MAP	>QL	p@5	>QL	MAP	>QL	p@5	>QL
QL	22.4	–	45.1	–	19.3	–	33.2	–
RelModel	28.9^l	**72.0**	50.7^l	34.0	24.1^l	67.0	38.4^l	31.0
CBRSD	$\mathbf{29.3}^l$	68.0	49.3	30.0	24.4^l	60.0	38.2^l	31.0
drift(clust)	$\mathbf{29.3}^l$	70.0	$\mathbf{53.1}^l_c$	**36.0**	$\mathbf{24.9}^l_r$	**68.0**	$\mathbf{40.8}^l_r$	**37.0**

	WSJ				ROBUST			
	MAP	>QL	p@5	>QL	MAP	>QL	p@5	>QL
QL	32.7	–	55.6	–	25.5	–	48.2	–
RelModel	38.7^l	**72.0**	**59.2**	**34.0**	27.6^l	61.2	46.9	25.2
CBRSD	$\mathbf{39.9}^l$	70.0	58.4	32.0	$\mathbf{30.7}^l$	**63.6**	**50.0**	**30.0**
drift(clust)	39.2^l	**72.0**	58.0	32.0	27.7^l_c	57.6	47.9	28.4

query, which is measured by the drift measure, seems to be a relatively effective estimate for the "faithfulness" of the relevance model to a presumed underlying information need. This finding is in accordance with a previous report about using drift to select a single relevance model from a set of models [17].

We can also see in Table 1 that all cluster-based implementations yield performance that is better in a majority of the relevant comparisons than that of RelModel, which constructs a single relevance model from all documents in $\mathcal{D}_{init}$. While the performance differences are, in general, not to a large scale, drift(clust), our best-performing method, outperforms RelModel to a statistically significant degree over SJMN for both MAP and p@5; also, there is only a single case (p@5 for WSJ) in which drift(clust) is outperformed by RelModel and the difference is not statistically significant. Recall that the performance of RelModel was optimized with respect to three free parameters, while that of our multiple-relevance-models was not optimized (except for the general choice of document-sets of size 10 for all implementations over all corpora). Thus, we view these results as gratifying, especially in light of the fact that parameters such as the number of terms are known to have considerable impact on the relevance-model performance. Furthermore, we note that RelModel can be used in Eq. 2 as one of the relevance models. Indeed, initial experiments with such implementation attest to the potential performance merits.

Performance Robustness. The relevance model, as other pseudo-feedback-based methods, suffers from a performance *robustness* problem [12,13]: for some queries the performance is worse than that of using only the original query (i.e., the QL method). Recent work [13] addresses this issue by constructing a relevance model using cluster-based document re-sampling (**CBRSD**) from $\mathcal{D}_{init}$ so as to "emphasize" documents with presumably high chances of relevance. We compare CBRSD — with re-sampling employed over the entire list $\mathcal{D}_{init}$ and the free

Table 3. Integrating multiple relevance models (our approach) vs. selecting a *single* (S-) relevance model [17] based on faithfulness measures. Boldface: best result in a column; 'l', 'r': statistically significant differences with QL and RelModel, respectively.

	AP		SJMN		WSJ		ROBUST	
	MAP	p@5	MAP	p@5	MAP	p@5	MAP	p@5
QL	22.4	45.1	19.3	33.2	32.7	55.6	25.5	**48.2**
RelModel	28.9^l	50.7^l	24.1^l	38.4^l	38.7^l	59.2	27.6^l	46.9
S-constdoc(clust)	28.4^l	45.7	23.9^l	37.6^l	$\mathbf{40.0}^l$	**61.2**	$\mathbf{28.8}^l$	45.9
constdoc(clust)	$\mathbf{29.5}^l$	52.7^l	24.5^l	39.0^l	39.4^l	56.8	28.4^l	47.9
S-clarity(clust)	27.4^l_r	47.3	23.6^l	35.8	37.8^l	57.6	24.2^l_r	35.5^l_r
clarity(clust)	29.3^l	52.9^l	24.6^l	39.6^l	39.6^l	58.0	27.1^l_r	44.7
S-drift(clust)	27.2^l_r	41.4_r	23.3^l	35.4_r	33.0_r	53.2	25.8_r	40.3^l_r
drift(clust)	29.3^l	$\mathbf{53.1}^l$	$\mathbf{24.9}^l_r$	$\mathbf{40.8}^l_r$	39.2^l	58.0	27.7^l	47.9

parameters set to the same values as those in our models and in RelModel — and drift(clust) in Table 2. We also report for both MAP and p@5 the percentage of queries (denoted ">QL") for which the performance transcends that of the QL method (i.e., performance robustness).

As we can see in Table 2 the performance of drift(clust) is in general better than that of CBRSD on AP and SJMN, while the reverse holds for WSJ and ROBUST. In addition, in most relevant comparisons the performance of drift(clust) is more robust than that of CBRSD. Moreover, while drift(clust) is more robust than RelModel in a majority of the comparisons, CBRSD is less robust than RelModel in most comparisons. Thus, we see that our approach of using multiple relevance models can help to improve performance robustness.

Comparison with Model Selection. As mentioned above, the clarity and drift faithfulness measures were used in previous work to select a *single* relevance model from a set of relevance models constructed by using document sampling from the initial list $\mathcal{D}_{init}$ [17]. Hence, in Table 3 we compare this model-selection paradigm (rows denoted with S-) with our approach that uses faithfulness measures to integrate models. (The uniform faithfulness measure does not constitute a selection criterion and is therefore not presented.) We can see that in most cases selecting a single relevance model yields performance that is inferior to that of our approach, and to that of RelModel. Thus, we conclude that there is merit in integrating multiple relevance models over selecting a single one.

5 Conclusion

We presented a novel probabilistic approach to ad hoc retrieval that *explicitly* addresses the uncertainty about the information need underlying a query. Our derived method integrates multiple relevance models by using their estimated *faithfulness* to the presumed information need. Empirical evaluation demonstrated the merits of our approach.

Acknowledgments. We thank the reviewers for helpful comments. This paper is based upon work supported in part by Google's and IBM's faculty research awards. Any opinions, findings and conclusions or recommendations expressed are those of the authors and do not necessarily reflect those of the sponsors.

References

1. Lavrenko, V., Croft, W.B.: Relevance-based language models. In: Proceedings of SIGIR, pp. 120–127 (2001)
2. Lavrenko, V.: A Generative Theory of Relevance, PhD thesis. University of Massachusetts Amherst (2004)
3. Zhai, C., Lafferty, J.D.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: Proceedings of SIGIR, pp. 334–342 (2001)
4. Li, X., Croft, W.B.: Improving the robustness of relevance-based language models. Technical Report IR-401, Center for Intelligent Information Retrieval. University of Massachusetts (2005)
5. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Precision prediction based on ranked list coherence. Information Retrieval 9(6), 723–755 (2006)
6. Lavrenko, V., Choquette, M., Croft, W.B.: Cross-lingual relevance models. In: Proceedings of SIGIR, pp. 175–182 (2002)
7. Liu, X., Croft, W.B.: Passage retrieval based on language models. In: Proceedings of CIKM, pp. 375–382 (2002)
8. Liu, X., Croft, W.B.: Cluster-based retrieval using language models. In: Proceedings of SIGIR, pp. 186–193 (2004)
9. Abdul-Jaleel, N., Allan, J., Croft, W.B., Diaz, F., Larkey, L., Li, X., Smucker, M.D., Wade, C.: UMASS at TREC 2004 — novelty and hard. In: Proceedings of TREC-13, pp. 715–725 (2004)
10. Zhai, C., Lafferty, J.D.: Model-based feedback in the language modeling approach to information retrieval. In: Proceedings of CIKM, pp. 403–410 (2001)
11. Mitra, M., Singhal, A., Buckley, C.: Improving automatic query expansion. In: Proceedings of SIGIR, pp. 206–214 (1998)
12. Collins-Thompson, K., Callan, J.: Estimation and use of uncertainty in pseudo-relevance feedback. In: Proceedings of SIGIR, pp. 303–310 (2007)
13. Lee, K.S., Croft, W.B., Allan, J.: A cluster-based resampling method for pseudo-relevance feedback. In: Proceedings of SIGIR, pp. 235–242 (2008)
14. Lavrenko, V., Croft, W.B.: Relevance models in information retrieval. In: [22], pp. 11–56.
15. Liu, X., Croft, W.B.: Evaluating text representations for retrieval of the best group of documents. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 454–462. Springer, Heidelberg (2008)
16. Lafferty, J.D., Zhai, C.: Document language models, query models, and risk minimization for information retrieval. In: Proceedings of SIGIR, pp. 111–119 (2001)
17. Winaver, M., Kurland, O., Domshlak, C.: Towards robust query expansion: Model selection in the language model framework to retrieval. In: Proceedings of SIGIR, pp. 729–730 (2007)

18. Saracevic, T., Kantor, P.: A study of information seeking and retrieving. iii. searchers, searches, and overlap. Journal of the American Society for Information Science 39(3), 197–216 (1988)
19. Belkin, N.J., Cool, C., Croft, W.B., Callan, J.P.: The effect of multiple query representations on information retrieval system performance. In: Proceedings of SIGIR, pp. 339–346 (1993)
20. Lee, J.H.: Analyses of multiple evidence combination. In: Proceedings of SIGIR, pp. 267–276 (1997)
21. Song, F., Croft, W.B.: A general language model for information retrieval (poster abstract). In: Proceedings of SIGIR, pp. 279–280 (1999)
22. Croft, W.B., Lafferty, J. (eds.): Language Modeling for Information Retrieval. Information Retrieval Book Series, vol. 13. Kluwer, Dordrecht (2003)

A Belief Model of Query Difficulty That Uses Subjective Logic

Christina Lioma[1], Roi Blanco[2], Raquel Mochales Palau[1], and Marie-Francine Moens[1]

[1] Computer Science, Katholieke Universiteit Leuven, 3000, Belgium
[2] IRLab, Computer Science Department, A Coruña University, Spain
christina.lioma@cs.kuleuven.be, rblanco@udc.es,
raquel.mochales-palau@cs.kuleuven.be, sien.moens@cs.kuleuven.be

Abstract. The difficulty of a user query can affect the performance of Information Retrieval (IR) systems. This work presents a formal model for quantifying and reasoning about query difficulty as follows: Query difficulty is considered to be a subjective belief, which is formulated on the basis of various types of evidence. This allows us to define a belief model and a set of operators for combining evidence of query difficulty. The belief model uses *subjective logic*, a type of probabilistic logic for modeling uncertainties. An application of this model with semantic and pragmatic evidence about 150 TREC queries illustrates the potential flexibility of this framework in expressing and combining evidence. To our knowledge, this is the first application of subjective logic to IR.

1 Introduction

The task of an Information Retrieval (IR) system is to retrieve information from a large repository of data in response to a user need, or *query*. The difficulty of this task may be affected by various factors, relating to the system or algorithms used, to the properties of the data to be retrieved, or to the inherent difficulty of the user's information need. The effect of the last of these factors upon retrieval performance is often referred to as *query difficulty*, and is studied extensively in the field (discussed in Section 5). Our work addresses query difficulty by proposing a formal framework for modelling query difficulty. Our proposed formalisation consists of a belief model that considers query difficulty to be a subjective belief, which is formulated on the basis of different types of evidence. This belief model uses a type of logic called subjective logic [11] in order to combine this evidence and to make a final estimation about the expected difficulty of a query. Any type of evidence can be used with this model.

Subjective logic is a type of probabilistic logic that allows probability values to be expressed with degrees of uncertainty. Like any probabilistic logic, it combines the strengths of logic and probabilities: from the area of logic, it draws the capacity to express structured argument models, and from the area of probabilities it draws the power to express degrees of those arguments. This means that one can reason with argument models in the presence of uncertain

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 92–103, 2009.

or partially incomplete evidence. Since most of our knowledge or evidence about query difficulty in IR can never be complete, but rather tends to include degrees of uncertainty, subjective logic constitutes an appealing model for representing query difficulty, in the sense that the conclusions drawn reflect any ignorance and uncertainty of the input evidence.

Subjective logic is not the only formalism to model degrees of uncertainty. Several other mathematical models have been proposed to this end, the oldest being the Bayesian model of subjective probabilities (a survey of its foundations can be found in [8]). There also exist generalisations of the Bayesian model, (critically surveyed in [21]), the best-known of which is Dempster-Shafer's *belief theory* [7,19]. The point of departure for Dempster-Shafer from classical Bayesian theory is its abandoning of the additivity principle of classical probabilities, i.e. the requirement that in a given event space, the probabilities of mutually disjoint elements must add up to 1. In classical Bayesian theory, this requirement makes it necessary to estimate a probability value for every element of the event space, even though there might be no basis for it, for instance in the case of uncertainty. Instead, Dempster-Shafer's belief theory suggests assigning a so-called *belief mass* to the whole event space. This belief mass is defined on the basis of both evidence and uncertainty about the event, hence it constitutes a much more flexible way of representing beliefs than traditional probabilities. Subjective logic can be seen as an alternative to the Dempster-Shafer theory, its main difference from the former being in its definition and distribution of belief mass: subjective logic defines belief mass as a function of not only belief and uncertainty, but also of an apriori probability in the absence of any evidence; furthermore, subjective logic assigns this belief mass, not to the whole event space, but to the individual elements of the event space. It can be argued that this allows subjective logic to formulate more expressive beliefs than Dempster-Shafer theory [11].

One of the advantages of using a belief theory, be it with Dempster-Shafer theory or subjective logic, is that it allows to operate on the beliefs and fuse them. Fusing beliefs is a formal way of saying 'combining evidence'. In the context of IR, combining evidence is a process that aims to use different types of information that may enhance IR performance, but for which we have different degrees of uncertainty regarding the enhancement that they may bring [18]. In this work we use two different subjective logic operations to combine evidence about query difficulty: a fair *consensus* and a biased *recommendation* (also called *discounting*).

The contribution of this work lies in proposing a type of formal logic for IR, which has not been used before in the field, and in illustrating its application to the representation and combination of evidence about query difficulty. To our knowledge, whereas Dempster-Shafer theory has been used extensively in IR (see Section 5), this is the first application of subjective logic to IR.

In the rest of this paper, Section 2 introduces belief models for subjective logic and presents our proposed belief model of query difficulty. Section 3 introduces the subjective logic operators for combining evidence used in this work. Section 4 illustrates the application of our proposed belief model of query difficulty with 150 TREC queries and different types of semantic and pragmatic evidence. Section 5 overviews related past work on logic models for IR and query difficulty. Section 6 summarises this work and suggests future research directions.

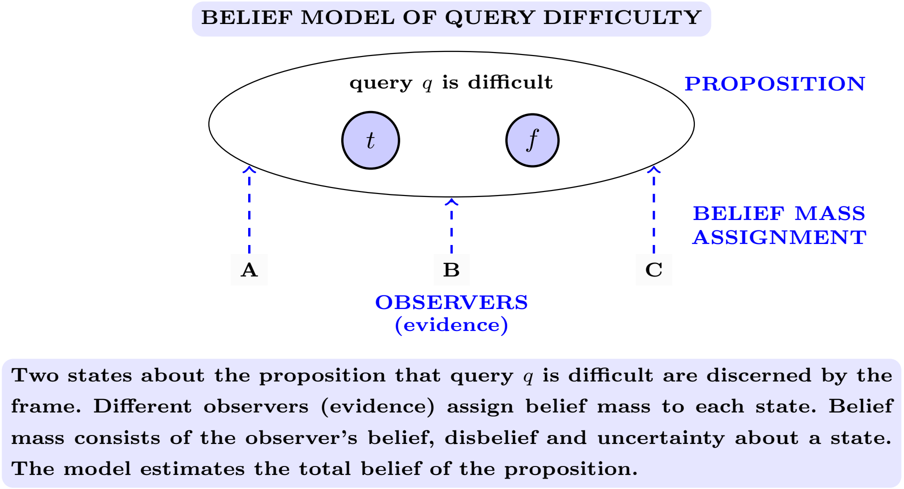
2 Belief Model of Query Difficulty

Belief models define a set of possible situations, for instance a set of possible states of a given system, called *frame of discernment*. This frame is defined over a proposition, i.e. a statement. At any one time, only one state of the frame of discernment can be true with respect to the proposition. A frame of discernment with two states ϕ and $\neg\phi$ is called *focused frame of discernment with focus on* ϕ. In this work, we use a focused frame of discernment.

Given any frame of discernment over a proposition, one can estimate the probability expectation that this proposition is true. This probability expectation is computed using evidence, which is said to come from 'observers'. An observer can assign to a state a *belief mass*, which represents his belief that this state is true with respect to the proposition. This belief can be represented in different ways by different uncertainty theories, for instance Dempster-Shafer or subjective logic as discussed in Section 1. An underlying similarity in these different representations is that this belief includes an explicit representation of the uncertainty of the observer about his belief.

Subjective logic considers the belief of an observer about the truth of a proposition as a subjective belief marked by degrees of uncertainty, and it calls it *opinion*. Let $\Phi = \{\phi, \neg\phi\}$ be a binary frame. An opinion about the truth of state ϕ is the ordered quadruple $\omega_\phi^A \equiv (b, d, u, a)$ where superscript A is the opinion's owner (i.e the **observer**), b is the belief mass supporting that the specified proposition is true (i.e. the **observer's belief**), d is the belief mass supporting that the specified proposition is false (i.e. the **observer's disbelief**), u is the amount of uncommitted belief mass (i.e. the **observer's uncertainty**), and a is the apriori probability in the absence of committed belief mass (divided uniformly among the states). These components satisfy: $b+d+u=1$ and $b, d, u, a \in [0,1]$. Clearly, a binomial opinion where $b+d=1$ is equivalent to a traditional probability, and a binomial opinion where $b+d=0$ expresses total uncertainty. The probability expectation of a binomial opinion is: $E = b + au$.

For the purpose of believing a binary proposition such as: "query q is difficult", we assume that the proposition will be either true or false. Hence, we define a focused frame of discernment as shown in Fig. 1 with the states: t (true) and f (false). The uncertainty probability of each state is represented by the belief mass assigned to each state by different observers, who are in fact our sources of evidence about query difficulty. The opinions of the three observers A, B, C shown in Fig. 1 are: $\omega_t^A \equiv (b_t^A, d_t^A, u_t^A, a_t^A)$, $\omega_f^A \equiv (b_f^A, d_f^A, u_f^A, a_f^A)$, $\omega_t^B \equiv (b_t^B, d_t^B, u_t^B, a_t^B)$, $\omega_f^B \equiv (b_f^B, d_f^B, u_f^B, a_f^B)$, $\omega_t^C \equiv (b_t^C, d_t^C, u_t^C, a_t^C)$, $\omega_f^C \equiv (b_f^C, d_f^C, u_f^C, a_f^C)$, where



Two states about the proposition that query q is difficult are discerned by the frame. Different observers (evidence) assign belief mass to each state. Belief mass consists of the observer's belief, disbelief and uncertainty about a state. The model estimates the total belief of the proposition.

Fig. 1. A belief model of query difficulty

subscripts t, f denote the true and false state respectively. These opinions define our **opinion space**. The observers' opinions are drawn from real observations about the queries, which constitute our **evidence space**, discussed next.

2.1 Evidence Space

For a focused frame of discernment, such as the one in Fig. 1, the proposition of the frame constitutes a binary event, where either the one or the other state is true: the query is either difficult or not. The type of evidence that we use to estimate the truth of this proposition can also be seen as binary, in the sense that it can be either positive (supporting that the query is difficult) or negative (supporting that the query is not difficult). Hence, both our opinion space and our evidence space consist of binary events. For such binary events, subjective logic defines a bijective mapping between the opinion and evidence space, as follows [11]. Let r denote positive evidence, and s denote negative evidence. Then, the correspondence between this evidence and the belief, disbelief, and uncertainty b, d, u is defined as:

$$b = \frac{r}{r+s+2} \qquad d = \frac{s}{r+s+2} \qquad u = \frac{2}{r+s+2} \tag{1}$$

Eq. 1 allows one to produce opinions based on statistical evidence. This mapping is derived in a mathematically elegant way, by considering the posteriori probability of the binary events defined in a focused frame of discernment, expressed using a beta probability function (the full derivation is presented in [11] and is outwith the focus of our work). The point to remember here is that in subjective logic any opinion has an equivalent mathematical and interpretative representation as a probability density function and vice versa.

3 Subjective Logic Operations for Combining Evidence

Subjective logic contains several operators for combining evidence (see [11] for more). In this work we use two combinations only: *consensus* and *recommendation* (or *discounting*). Using subjective logic terminology, we will refer to combining evidence as combining opinions, and treat these statements as equivalent.

3.1 Consensus between Independent Opinions

Let $\omega_x^A \equiv (b_x^A, d_x^A, u_x^A, a_x^A)$ and $\omega_x^B \equiv (b_x^B, d_x^B, u_x^B, a_x^B)$ be opinions respectively held by two independent observers A and B about the same proposition x. Then, $\omega_x^{A,B} \equiv (b_x^{A,B}, d_x^{A,B}, u_x^{A,B}, u_x^{A,B})$ is the opinion of an imaginary observer $[A, B]$ about x. $[A, B]$ represents the *Bayesian consensus* of opinions of both A and B, denoted $\omega_x^{A,B} = \omega_x^A \oplus \omega_x^b$, and defined by:

$$b_x^{A,B} = \frac{b_x^A u_x^B + b_x^B u_x^A}{\kappa}, \qquad d_x^{A,B} = \frac{d_x^A u_x^B + d_x^B u_x^A}{\kappa}, \qquad u_x^{A,B} = \frac{u_x^A u_x^B}{\kappa} \tag{2}$$

$$a_x^{A,B} = \frac{a_x^B u_x^A + a_x^A u_x^B - (a_x^A + a_x^B) u_x^A u_x^B}{u_x^A + u_x^B - 2 u_x^A u_x^B} \tag{3}$$

where $\kappa = u_x^A + u_x^B - u_x^A u_x^B$ such that $\kappa \neq 0$, and where $a_x^{A,B} = (a_x^A + a_x^B)/2$ when $u_x^A, u_x^B = 1$. The proof is included in [11].

This operation is both commutative and associative, meaning that the order in which opinions are combined does not impact the combination. The operation assumes that opinions are independent and that not all the combined opinions have zero uncertainty. Attempting to combine opinions all of which have zero uncertainty can be seen as meaningless, because these opinions would have complete belief or disbelief, and would hence be in complete conflict or agreement.

The effect of the consensus operator is to reduce uncertainty. The consensus operator has the same purpose as Dempster's rule [7], and the two tend to produce overall quite similar results. In [11], Section 5.3, Josang illustrates some cases where the consensus operator is 'better' than Dempster's rule, in the sense that the former produces less counter-intuitive results than the latter.

3.2 Recommendation (or Discounting) between Opinions

Assume two observers A and B where A has an opinion about B, and B has an opinion about a proposition x. A recommendation of these two opinions consists of combining A's opinion about B with B's opinion about x[1] in order for A to get an opinion about x. Let $\omega_x^B \equiv (b_x^B, d_x^B, u_x^B, a_x^B)$ be B's opinion about x expressed in a recommendation to A, and let $\omega_B^A \equiv (b_B^A, d_B^A, u_B^A, a_B^A)$ be A's opinion about

[1] B's recommendation must be interpreted as what B recommends to A, and not necessarily as B's real opinion.

B's recommendation. Then, $\omega_x^{AB} = \omega_B^A \otimes \omega_x^B$ is A's opinion about x as a result of the recommendation from B, defined as:

$$b_x^{AB} = b_B^A b_x^B, \qquad d_x^{AB} = b_B^A d_x^B, \qquad u_x^{AB} = d_B^A + u_B^A + b_B^A u_x^B \qquad a_x^{AB} = a_x^B \quad (4)$$

This operation is associative but not commutative, meaning that the order in which opinions are combined impacts the combination. Eq. 4 can become equivalent to Shafer's discounting function [19] by setting $1 - c = b_B^A$, where c denotes Shafer's discounting rate which is multiplied to the belief mass on each state in the frame except the belief mass of the power-set itself.

4 Illustrative Experiments

4.1 Evidence of Query Difficulty

We illustrate an application of our formalisation of query difficulty using two types of linguistic evidence, namely semantic and pragmatic evidence. From these types of linguistic evidence we obtain positive and negative evidence of query difficulty, which we map into belief, disbelief and uncertainty using Eq. 1.

The choice of evidence is illustrative. Our proposed model allows to represent and combine any type of evidence, simply by introducing more observers who contribute their beliefs to the frame of discernment. Any other evidence can be used.

Semantic Evidence. We use as semantic evidence two indicators that have been found to be correlated with query difficulty, namely (i) query scope, proposed by Plachouras and Ounis [18], and (ii) query polysemy [16]. Query scope is a probabilistic measure that estimates how specific or generic a query is by using the query term frequencies in the collection as well as their semantic content. Assuming that each query term corresponds to one or more concepts in WordNet (or any other similar lexical reference system), the semantic content of query terms is approximated from the hierarchical structure of their respective concepts. Specifically, we use the following formulae from [18] (keeping their original notation): given a term t_k and several concepts $\mathbf{C}_k$ associated to this term, the scope of t_k is defined as the maximum probability of any of its associated concepts: $scope_{t_k} = \max_{C \in \mathbf{C}_k} prob(C)$. This probability is estimated as follows: $prob(C) = \sum_{C = C_{k,j} \in \mathbf{C}_k} a_{k,j} \cdot \frac{tf_k}{T}$, where $C_{k,j}$ denotes the j^{th} concept (among all $\mathbf{C}_k$) associated to t_k, $a_{k,j}$ denotes the 'contribution' of t_k to concept $C_{k,j}$, tf_k is the frequency of t_k in the collection, and T is the sum of all the frequencies of all terms in the collection. The contribution $a_{k,j}$ is defined as: $a_{k,j} = \frac{(D_k+1) - d_{k,j}}{n_k(D_k+1) - \sum_{j=1}^{n_k} d_{k,j}}$, where $d_{k,j}$ denotes the length of the path from concept $C_{k,j}$ to the most generic concept in the WordNet hierarchy, D_k denotes the maximum path length of concepts $C_{k,j} \in \mathbf{C}_k$, and n_k denotes the number of concepts associated with t_k (in this work we select n_k among $C_{k,j}$ only). Plachouras and Ounis posit that as term scope approaches zero, the term is less represented in the collection, and hence more difficult to retrieve [18]. Based on this reasoning, we define a threshold θ_{sco}, so that any term scope $\leq \theta_{sco}$ constitutes positive evidence of query difficulty, and any term score $> \theta_{sco}$ constitutes negative evidence. For the illustrations shown here, we define θ_{sco} as the median term scope in all queries.

The second type of semantic evidence consists of the 'polysemy score' offered by WordNet to each term. This score reflects the number of concepts to which a term is associated, e.g. a score of 1 denotes a monosemous term. WordNet considers terms of polysemy score 1-4 as uncommon (in decreasing degrees from 1 to 4) and terms of polysemy score 5 or more as common (in increasing degrees from 5 upwards). Following the assumption that the more polysemous a term is, the harder the query [16], we define the following threshold: $\theta_{pol} \leq 4$ constitutes positive evidence of query difficulty, and $\theta_{pol} > 5$ constitutes negative evidence. Here, the value of the threshold θ_{pol} is taken directly from WordNet.

Pragmatic Evidence. Our pragmatic evidence aims to show whether a query constitutes a *literal* or *stipulative* statement of an information need. The meaning of a literal statement remains unchanged in all contexts, whereas the meaning of a stipulative statement is context- or register-dependent. We assume that a literal query should be easier to retrieve than a stipulative query because its meaning depends more on the literal semantics of its individual terms, and less on their contextual, metaphorical or other interpretations. To obtain this evidence, we use human judges, who read the queries and classify them as stipulative or not, based on their intuition. We use three human judges and consider their decision about the query being a literal or stipulative statement of an information need as negative or positive evidence of query difficulty respectively. The judges have a disagreement rate of 17.1%, and an inter-annotator agreement of $\kappa = 0.413$, measured using Cohen's κ, which indicates moderate agreement.

To recapitulate, we have three types of evidence (scope, polysemy, pragmatic judgement), which correspond to the observers of our model (Fig. 1). Next we illustrate how we formalise this evidence in our belief model of query difficulty, using TREC [24] queries.

4.2 Working Examples

Let us consider the following TREC queries: n^o 415 (`drugs, Golden Triangle`), n^o 479 (`where can I find information about kappa alpha psi?`), n^o 492 (`us savings bonds`), n^o 508 (`hair loss is a symptom of what diseases?`). In this section, we will estimate their difficulty and compare it to the retrieval performance they yield on their respective TREC collections (LAT & WT10G). Retrieval will be realised with the BM25 model at default settings, and measured by Mean Average Precision (MAP) against the prejudged relevance information provided for these datasets by TREC.

Table 1 presents the evidence given by our observers about the difficulty of each sample query, as well as the respective belief mass estimated by our model. The opinion of each observer can be represented as a tuple of the belief mass (b, d, u) and also of a prior probability of uncertainty a (a is divided uniformly

Table 1. Sample queries with their respective query difficulty evidence (scope, polysemy, pragmatic judgement) and MAP. The belief mass components of each type of evidence are clearly shown (b, d, u), as well as their final expected probability of query difficulty (E)

Proposition: ***the query is difficult***													
	query scope				**query polysemy**				**pragmatic judgement**				
query	belief	disbef.	uncert.	Exp.diff	belief	disbef.	uncert.	Exp.diff	belief	disbef.	uncert.	Exp.diff	MAP
	b	d	u	E	b	d	u	E	b	d	u	E	
415	0.13	0.63	0.25	0.25	0.40	0.20	0.40	0.50	0.60	0.00	0.40	0.80	0.250
479	0.50	0.33	0.17	0.58	0.29	0.43	0.29	0.44	0.60	0.00	0.40	0.80	0.240
492	0.75	0.13	0.13	0.88	0.20	0.40	0.40	0.30	0.60	0.00	0.40	0.80	0.307
508	0.79	0.11	0.11	0.84	0.33	0.33	0.33	0.50	0.00	0.60	0.40	0.20	0.230

Table 2. Two different combinations of semantic scope (sco), polysemy (pol) and pragmatic (pra) evidence about the four sample queries: consensus combines all evidence fairly, whereas discounting favours pragmatic evidence at the expense of the other two

Combination of evidence								
	consensus sco,pol,pra				**pra discounts sco,pol**			
query	belief	disbef.	uncert.	Exp.diff	belief	disbef.	uncert.	Exp.diff
415	0.15	0.36	0.50	0.394	0.54	0.00	0.46	0.770
479	0.27	0.23	0.50	0.524	0.54	0.00	0.46	0.770
492	0.32	0.19	0.44	0.569	0.54	0.00	0.46	0.770
508	0.31	0.19	0.50	0.558	0.00	0.54	0.46	0.230

among the two states of our frame of discernment, hence a=0.5 at all times). For example, the opinion of the polysemy observer that query 492 is difficult is represented as $\omega_t^{pol} \equiv (0.4, 0.2, 0.4, 0.5)$. In this case, the observer's belief is equal to his uncertainty (=0.4), and his disbelief is low (=0.2), hence the evidence of this observer for this query is not very discriminative.

In Table 1 we also see that the different types of evidence do not always agree. For instance, using query scope evidence, the probability that query 415 is difficult is quite low (0.25), whereas using pragmatic evidence, the same probability for the same query is quite high (0.80). How difficult this query is can be seen in the last column of the table, which presents the MAP obtained by the system for this query. In this case, an MAP of 0.250 is relatively low, hence the pragmatic evidence seems more appropriate than the semantic scope evidence.

A more accurate estimate of query difficulty could be obtained by combining those different types of evidence, as presented in Table 2. Table 2 illustrates the two combinations of evidence presented in Section 3. The column headed 'consensus' refers to the combination of our semantic scope, polysemy and pragmatic evidence (denoted 'sco,pol,pra' respectively) using Eq. 2. The column headed 'pra discounts sco,pol' refers to the combination by discounting (Eq. 4), where pragmatic evidence recommends its opinion to the consensus of scope and polysemy evidence. In this case, more weight is given to the pragmatic evidence than to the other two types of evidence. We see that combining evidence by consensus provides probability estimates that constitute a fair compromise of the individual expectations of each type of evidence. However, this is not always desirable, especially in cases such as the ones presented in Table 2, where the original estimates were in sharp disagreement between them. Combining strongly disagreeing evidence results in an expected probability that approaches 0.5, hence which can be considered relatively arbitrary. This implies that combining evidence by consensus may be better suited to generally agreeing evidence, than to sharply disagreeing evidence. On the contrary, combining evidence by discounting allows one to produce more biased estimates (in this context, bias is desirable). A prerequisite for such cases would be having some apriori knowledge regarding the reliability or suitability of each type of evidence, or about the agreement between the types of evidence to be combined. In a realistic situation, this type of knowledge is not difficult to obtain, since most systems that use query difficulty evidence for retrieval prediction ensure such knowledge using offline training and pre- or post-retrieval passes on prejudged relevant datasets (evidence relying on such processes in highlighted in Section 5). We see in Table 2 that combination by discounting produces estimates that are more discriminative than the consensus estimates, namely 0.77 and 0.23 as opposed to estimates closely approaching 0.5. The 0.77 estimates are in fact more accurate predictions of query difficulty, because the displayed queries are difficult queries (their MAP scores do not exceed 0.3, as shown in Table 1).

Finally, we can report that the observations reported illustratively above are also valid for the majority of the 401-550 TREC query set. Experiments with these 150 queries show that semantic scope is not discriminative evidence of query difficulty, that polysemy is better than scope but not at all times, and that pragmatic judgement constitutes the most reliable out of the three sources of evidence. More importantly, the combination of evidence for all queries is consistently better when we use discounting (with pragmatic evidence discounting the other two), than when we combine all three types of evidence on equal grounds with consensus. The respective correlation between the estimated query difficulty and MAP is in the range of Spearman's $\rho \approx 0.3$ for discounting (weak positive correlation), and $\rho \approx 0.1$ for consensus. These correlations are not strong, as is commonly reported for most types of query difficulty evidence, and in particular evidence stemming from the textual expression of the query [16]. These weak correlations are partly due to the reliability or quality of the evidence used, but also to the fact that the problem of query difficulty is largely influenced by several factors (as mentioned in Section 1), meaning that it is practically impossible for a single type of evidence to constitute a reliable and consistent predictor of query difficulty for all queries in all datasets [18].

5 Related Work

In order to avoid breaking the flow of the belief model presented in this work, we have left the treatment of related work at the end. This section discusses separately applications of formal logic to IR, and work on query difficulty. The applications of formal logic aim to give a plenary view of the different aspects and processes of an IR system that can be formalised with logic, hence constituting potential future applications of the subjective logic presented in this work. The overview of work on query difficulty aims to present different types of query difficulty evidence, which can be used within our proposed frame, using the same methodology and equations set out in Sections 2-3.

Formal Logic in IR. The expressive power of formal logic has long attracted applications of it to IR, starting with Van Rijsbergen's *logical uncertainty principle* [1]. Since then, modal logic has been used to integrate semantic-based and probabilistic-based approaches of deciding the relevance between a document and a query [17]. Extensions of the logical uncertainty principle have been proposed in order to integrate natural language processing and artificial intelligence techniques to IR [5]. Particular aspects of formal logic have also been used to address specific aspects or processes in IR, for instance belief revision has been used to model IR agents [14], to estimate the similarity between a document and a query [15], and more recently to model adaptive and context-sensitive IR [13]. The Dempster-Shafer theory presented in Section 1 has been used extensively: to build an IR framework where information structure, significance, uncertainty and partiality can be elegantly represented and processed [12], to integrate Web evidence into IR [23], to integrate into Web IR evidence of query difficulty in the form of semantic scope (one of the types of evidence we used in this work) [18], as well as to relate dependent indices [20]. There are further applications of formal logic to IR, reviews of which can be found in [3]. A more in-depth treatment of formal representations for IR can be found in [2].

Query Difficulty. In this work we propose a formal representation of query difficulty, an area of much interest to IR. Difficult queries may be due to a number of causes. Linguistic features of the query text that may indicate query difficulty include morphological statistics (e.g. word length, number of morphemes per word), syntactical statistics (e.g. number of conjunctions, syntactic depth), or semantic statistics (e.g. polysemy value) [16]. Additional factors that may impact retrieval performance can be drawn from the retrieval resources. For instance, simple statistics such as the frequency of query terms in the collection [10], or the score of the top-ranked documents and the average inverse document frequency (idf) of query terms [22] have been correlated to query difficulty. Query difficulty has also been correlated with query length [25], based on the overlap between results of sub-queries based on single query terms and results of longer queries. A *clarity score* has been proposed [6] to measure the coherence of a list of retrieved documents by the KL-divergence between the query model and the collection model. A *robustness score* [26] has been proposed to quantify the robustness of the document ranking in the presence of uncertainty. Retrieval precision has been correlated to the distance between the retrieved document set and the collection [4] measured by the Jensen-Shannon divergence. In addition, different techniques have been proposed for predicting automatically query performance specifically in Web IR [27], either by making use of both single term and term proximity features to estimate the quality of top retrieved documents, or by viewing the retrieval system as a noisy channel, where the query is the input, the ranked list of documents is the corrupted output, and their proposed technique measures the degree of corruption. The main components of query difficulty have been defined as the textual expression of the query, the set of documents relevant to the query and the entire collection of documents, with experiments showing that query difficulty strongly depends on the distances between these components [4]. Finally, a recent overview of query difficulty with respect to performance prediction can be found in [9].

6 Conclusion

We proposed representing and formalising query difficulty for IR using subjective logic, a type of probabilistic logic for modelling uncertainties not used in IR before. Considering query difficulty as a subjective belief, formulated on the basis of various types of evidence, we defined a belief model that uses subjective logic, and a set of operators for combining evidence of query difficulty. We illustrated an application of this model with semantic and pragmatic evidence and TREC queries, which were combined in two different ways: by fair consensus and by biased discounting. Integrating further evidence or refining its combination can be realised easily with subjective logic, as illustrated in this work with working examples. Further research includes obtaining more varied sources of evidence for the task of query difficulty (any of the types of evidence highlighted in Section 5 can be used). Finally, the proposed belief model could be applied to other aspects of IR, apart from query difficulty, similarly to the varied and extensive use of Dempster-Shafer by the community (overviewed in Section 5).

Acknowledgements. Author 2 was partially supported by *Ministerio de Ciencia e Innovación*, FEDER and *Xunta de Galicia* under projects TIN2008-06566-C04-04 and 07SIN005206PR

References

1. van Rijsbergen, C.J.: A non-classical logic for information retrieval. Comput. J. 29(6), 481–485 (1986)
2. van Rijsbergen, C.J.: The Geometry of Information Retrieval. CUP, Cambridge (2004)
3. van Rijsbergen, C.J., Crestani, F., Lalmas, M.: Information Retrieval: Uncertainty and Logics. Springer, Heidelberg (1998)
4. Carmel, D., Yom-Tov, E., Darlow, A., Pelleg, D.: What makes a query difficult? In: SIGIR, pp. 390–397 (2006)
5. Chiaramella, Y., Chevallet, J.-P.: About retrieval models and logic. Comput. J. 35(3), 233–242 (1992)
6. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: SIGIR, pp. 299–306 (2002)

7. Dempster, A.P.: A generalization of Bayesian inference. Journal of the Royal Statistical Society B(30), 205–247 (1968)
8. Fishburn, P.C.: The axioms of subjective probability. Statistical Science 3(1), 335–345 (1986)
9. Hauff, C., Azzopardi, L., Hiemstra, D.: The combination and evaluation of query performance prediction methods. In: ECIR, pp. 301–312 (2009)
10. He, B., Ounis, I.: Inferring query performance using pre-retrieval predictors. In: Apostolico, A., Melucci, M. (eds.) SPIRE 2004. LNCS, vol. 3246, pp. 43–54. Springer, Heidelberg (2004)
11. Josang, A.: A logic for uncertain probabilities. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 9(3), 279–311 (2001)
12. Lalmas, M.: Information retrieval and Dempster-Shafer's theory of evidence. In: Hunter, A., Parsons, S. (eds.) Applications of Uncertainty Formalisms. LNCS (LNAI), vol. 1455, pp. 157–176. Springer, Heidelberg (1998)
13. Lau, R.Y.K., Bruza, P.D., Song, D.: Towards a belief-revision-based adaptive and context-sensitive information retrieval system. ACM Trans. Inf. Syst. 26(2) (2008)
14. Logan, B., Reece, S., Sparck Jones, K.: Modelling information retrieval agents with belief revision. In: SIGIR, pp. 91–100 (1994)
15. Losada, D.E., Barreiro, A.: A logical model for information retrieval based on propositional logic and belief revision. Comput. J. 44(5), 410–424 (2001)
16. Mothe, J., Tanguy, L.: Linguistic features to predict query difficulty - a case study on previous TREC campaigns. In: SIGIR Workshop on Predicting Query Difficulty: Methods and Applications (2005)
17. Nie, J.-Y.: Towards a probabilistic modal logic for semantic-based information retrieval. In: SIGIR, pp. 140–151 (1992)
18. Plachouras, V., Ounis, I.: Dempster-Shafer theory for a query-biased combination of evidence on the web. Inf. Retr. 8(2), 197–218 (2005)
19. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
20. Shi, L., Nie, J.-Y., Cao, G.: Relating dependent indexes using Dempster-Shafer theory. In: CIKM, pp. 429–438 (2008)
21. Smets, P.: What is Dempster-Shafer's model? Wiley, Chichester (1994)
22. Tomlinson, S.: Robust, web, and terabyte retrieval with Hummingbird SearchServer at TREC 2004. In: TREC (2004)
23. Tsikrika, T., Lalmas, M.: Combining evidence for web retrieval using the inference network model: an experimental study. Inf. Process. Manage. 40(5), 751–772 (2004)
24. Voorhees, E.M., Harman, D.K.: TREC: Experiment and Evaluation in Information Retrieval. MIT Press, Cambridge (2005)
25. Yom-Tov, E., Fine, S., Carmel, D., Darlow, A.: Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In: SIGIR, pp. 512–519 (2005)
26. Zhou, Y., Croft, W.B.: Ranking robustness: a novel framework to predict query performance. In: CIKM, pp. 567–574 (2006)
27. Zhou, Y., Croft, W.B.: Query performance prediction in web search environments. In: SIGIR, pp. 543–550 (2007)

"A term is known by the company it keeps": On Selecting a Good Expansion Set in Pseudo-Relevance Feedback

Raghavendra Udupa[1], Abhijit Bhole[2], and Pushpak Bhattacharyya[2]

[1] Microsoft Research India
Bangalore 560080
raghavu@microsoft.com
http://research.microsoft.com/en-us/labs/india
[2] Department of Computer Science and Engineering
Indian Institute of Technology, Bombay
Mumbai 400076
{abhijit.bhole,pb}@cse.iitb.ac.in
http://www.cse.iitb.ac.in/~pb

Abstract. It is well known that pseudo-relevance feedback (PRF) improves the retrieval performance of Information Retrieval (IR) systems in general. However, a recent study by Cao et al [3] has shown that a non-negligible fraction of expansion terms used by PRF algorithms are harmful to the retrieval. In other words, a PRF algorithm would be better off if it were to use only a subset of the feedback terms. The challenge then is to find a good expansion set from the set of all candidate expansion terms. A natural approach to solve the problem is to make term independence assumption and use one or more term selection criteria or a statistical classifier to identify good expansion terms independent of each other. In this work, we challenge this approach and show empirically that a feedback term is neither good nor bad in itself in general; the behavior of a term depends very much on other expansion terms. Our finding implies that a good expansion set can not be found by making term independence assumption in general. As a principled solution to the problem, we propose spectral partitioning of expansion terms using a specific term-term interaction matrix. We demonstrate on several test collections that expansion terms can be partitioned into two sets and the best of the two sets gives substantial improvements in retrieval performance over model-based feedback.

Keywords: Information Retrieval, Relevance Feedback, Pseudo-relevance Feedback, Expansion Terms, Term-Document Matrix.

1 Introduction

Pseudo-relevance feedback (PRF) is a well-known method for query expansion in Information Retrieval (IR) [1]. In general, PRF uses frequent terms in the top

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 104–115, 2009.

results of the first pass retrieval as expansion terms. The assumption underlying PRF is that the top-ranked documents contain terms related to the query terms and hence help identifying documents relevant to the query. Although conceptually simple, PRF is a very powerful technique for improving retrieval performance and is highly effective as a query expansion technique.

PRF techniques typically apply one or more criteria on the terms in the feedback documents and select terms that satisfy the criteria. Commonly employed criteria include term distributions in the feedback documents and the collection, idf, query length, and linguistic features [2]. It is common to assume that the selected expansion terms are all related to the query and use all of them in the second pass retrieval.

A recent study by Cao et al questioned the basic assumption of PRF and found that a non-negligible fraction of the expansion terms were actually harmful to the query[1] [3]. In other words, PRF would be better off if it were to use only a subset of the expansion terms. To find a good set of expansion terms, Cao et al used a SVM-based statistical classifier to select good terms. Their approach is crucially based on the assumption that *"an expansion term acts on the query independent of other terms"* and therefore, a good term can be identified independent of other expansion terms. Cao et al showed that their approach gives improvements in the retrieval performance over both model-based feedback [4] and relevance based language model [5].

In this work, we challenge the assumption that good terms can be identified independent of other expansion terms. We claim that a term is neither good nor bad in itself in general. The impact of including a term into the expansion set is a function of the term being added as well as other terms in the expansion set. The same term can behave in opposite ways depending on the company of other terms as far as retrieval performance is concerned. This is because terms interact with each other and as a consequence there is a portfolio effect. To give an analogy, if it is moderately cold, a jacket or a sweater and a shawl is more preferable than a sweater, a jacket, and a shawl together. Whereas the first two sets, i.e. {jacket} and {sweater, shawl}, are likely to keep you warm, their union will most probably make you feel uncomfortable. The effect of jacket is positive when used alone and negative when used along with sweater and shawl. In other words, the effect of jacket on your comfort depends on what other winter wear you are using along with it.

We provide solid empirical evidence for our claim: we show that in almost all topics from real test collections, a majority of the expansion terms behave inconsistently; they improve retrieval performance when paired with one set of expansion terms and degrade retrieval performance when paired with a different set.

An implication of our findings is that a good expansion set can not, in general, be discovered in a principled manner by approaches that choose terms independent of other terms. Such approaches make use of a flawed notion of goodness of

[1] Cao et al considered 150 topics from each of AP, WSJ, and Disk4&5 collections and 80 expansions with the largest probabilities for each topic. Approximately 50% of the terms were found to be neutral and 30% to be harmful.

expansion terms. **A principled solution to the problem of finding good expansion set must take into account interacting terms**. Such a solution will take a collective decision on all the expansion terms instead of independent decisions on individual terms.

We propose spectral partitioning of expansion terms as a principled solution for the problem of finding a good expansion set. Spectral partitioning takes into account interactions between terms and enables us to take a collective decision on all the expansion terms. In our partitioning experiments, we employ a weighted term-document matrix which implicitly defines a term-term interaction matrix. However, we may use any appropriately defined term-term interaction matrix in general. Given such a matrix, terms can be partitioned using standard techniques such as SVD or Graph Laplacian [6,7,8].

In the remainder of this paper we provide an exposition of our approach along with results of empirical investigations on multiple test collections. We start by discussing some of the important previous research work on PRF in Section 2. Next we re-examine the term independence assumption in Section 3. We describe our spectral partitioning based approach to PRF in Section 4. Next we discuss the experimental setup and results of our empirical investigations in Section 5. Finally, we discuss the results and propose some ideas for future investigation in Section 6.

2 Related Work

Pseudo-Relevance Feedback has a long history in IR [1,9]. It was first implemented in the vector space models [1] and subsequently has made its way into probabilistic models and language models [4,5]. Since our work, like that of Cao et al [3], is in the framework of language models, we restrict our discussion to the implementations of PRF in this framework only. For an insightful and thorough discussion on feedback in language models, please see Section 5 of the recent survey on language models [10].

In the language modeling framework, documents are ranked according to the negative Kullback-Leibler (KL) divergence [11] of the query language model θ_Q with the (smoothed) document language model θ_D.

$$Score\left(Q,D\right) = -D\left(\theta_Q \| \theta_D\right) \overset{rank}{=} \sum_{w \in V} P\left(w\left|\theta_Q\right.\right) \log P\left(w\left|\theta_D\right.\right) \tag{1}$$

It is in the re-estimation of the query model θ_Q that feedback information can be leveraged. In model-based feedback [4], the original query model θ_Q is interpolated with a feedback topic model θ_F estimated from the feedback documents from the first pass retrieval:

$$P\left(w\left|\theta_Q^{'}\right.\right) = \left(1-\alpha\right) P\left(w\left|\theta_Q\right.\right) + \alpha P\left(w\left|\theta_F\right.\right) \tag{2}$$

where is the interpolation parameter $\alpha \in [0,1]$ used to control the amount of feedback. There are several ways in which the topic model θ_F can be learnt from the feedback documents in practice [4,12]. One approach is to employ a two component mixture where one component is a fixed background model θ_C that explains the background words in the feedback documents and the other component is an unknown topic model θ_F that explains the topical words [4]. EM algorithm can be employed to estimate the topic model by maximizing the likelihood of the feedback documents. In our study, we used the feedback terms computed using this approach.

An alternative to model-based feedback is relevance-based language model, where the relevance model θ_R is estimated by assuming that the feedback documents are samples from the relevance model [5]. The original query model θ_Q is interpolated with θ_R in a manner analogous to model-based feedback. It should be noted that both model-based and relevance-based models use all the feedback terms for expansion. Whereas the topic model assigns higher probability mass to the most distinctive terms in the feedback documents, the relevance model assigns higher probability mass to the most frequent terms from the feedback documents. In contrast to both model-based feedback and relevance-based query expansion, the approach of Cao et al uses a subset of the feedback terms in expansion. They employ a statistical classifier for identifying good expansion terms [3].

3 Re-examination of the Independence Assumption

The main claim of our work is the following: **the effect of including a term into an expansion set on retrieval depends on the rest of the terms in the expansion set**. Before we go on to describe the experimental procedure for validating our claim, we discuss some motivating examples in Section 3.1.

3.1 Motivating Examples

As our first motivating example, we consider Topic 164 from TREC 3 (AP88-89): **Generic Drugs - Illegal Activities by Manufacturers**. The top 4 expansion terms of the topic model estimated from the top 10 feedback documents are **drug**, **generic**, **fda**, and **compani**. Now, consider **subcommitte** and **result**, two candidate expansion terms for this topic. According to term goodness criterion of Cao et al (see Section 3 of [3]), which makes independence assumption, **subcommitte** is a bad term whereas **result** is a good term. However, they behave very differently with different expansion sets. For instance, **subcommitte** acts as a good term when used with the set {**drug**, **generic**} and when used with {**drug**, **generic**, **fda**}, it acts as a bad term. Similarly, **result** acts as a good term with {**drug**, **generic**, **fda**} and becomes a bad term when used with {**drug**, **generic**, **fda**, **compani**}.

As our second example, we consider the title of Topic 308 from TREC 6 (Disk4&5): **Implant Dentistry**. The top 4 expansion terms of the feedback topic model are **devic**, **implant**, **chiropract**, and **fda**. Consider **prosthesi** and **requir**, two candidate expansion terms for this topic. According to term goodness criterion of Cao et al **prosthesi** is a good term whereas **requir** is a bad term. However, **prosthesi** acts as a good term when used with the set {**devic**, **implant**} and when used with {**devic**, **implant**, **chiropract**}, it acts as a bad term. On the other hand, **requir** acts as a bad term when used with the set {**devic**, **implant**, **chiropract**} and as a good term when used with {**devic**, **implant**, **chiropract**, **fda**}.

Continuing the investigation of the above topics, we did the following experiment: For each feedback term t, we checked the effect of adding t to each of the sets $T_1, \ldots, T_{25}$, where T_k is the set of top k terms of the feedback topic model[2]. Our aim was to find out how many feedback terms behave consistently with respect to the sets $T_1, \ldots, T_{25}$. If the term independence assumption were indeed valid then most terms must behave consistently. However, our investigation revealed that 56% of the feedback terms of the query **Generic Drugs - Illegal Activities by Manufacturers** (Topic 164, TREC 3) are inconsistent. In the case of the query **Implant Dentistry** (Topic 308, TREC 6), the percentage of inconsistent feedback terms was even higher, 92%.

Figure 1 shows the behavior of the term **prosthesi** (Topic 308, TREC 6) when it is used with $T_1, \ldots, T_{25}$. As can be seen from the figure, the behavior of the term is highly inconsistent across the expansion sets. It acts as good, bad, or neutral depending on the expansion set with which it is used. These examples not only show that terms are neither good nor bad in isolation but also suggest that term selection strategies that make independence assumption must not be trusted in general.

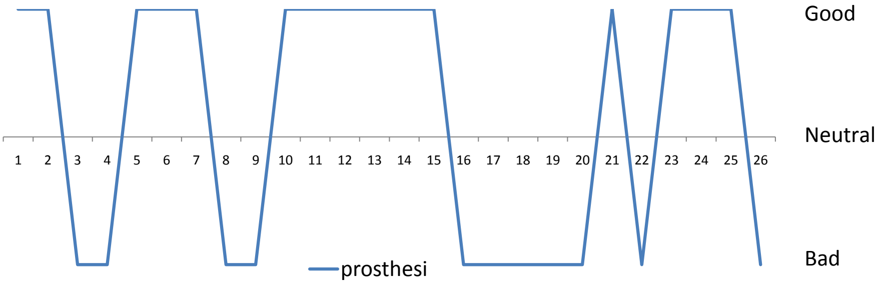


Fig. 1. The inconsistent behavior of **prosthesi** when used with different expansion sets

3.2 Empirical Validation of the Term Dependence Claim

In this section, we describe the experiments we did on topics from several collections to determine the set of inconsistent terms (i.e. terms which behave differently with different expansion sets) from each topic.

Model. Let Q be a topic and θ_F be the feedback topic model estimated using the two component mixture model approach described in Section 2. Let S_F be

[2] We ranked the feedback terms according to $P(w|\theta_F)$.

the set of top 100 terms in θ_F and $S \subseteq S_F$ be an expansion set. We formed a new feedback topic model θ_F^S as follows:

$$P\left(w\left|\theta_F^S\right.\right) = \begin{cases} \dfrac{P\left(w\left|\theta_F\right.\right)}{\sum_{w' \in S} P\left(w'\left|\theta_F\right.\right)} \text{ if } w \in S \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

The re-estimated query model is then $P\left(w\left|\theta_Q^{'}\right.\right) = (1-\alpha)\,P(w\left|\theta_Q\right.) + \alpha P\left(w\left|\theta_F^S\right.\right)$. To integrate a new term t to θ_F^S, we used the following scoring function:

$$Score\left(Q, D\right) = \sum_{w \in V} P\left(w\left|\theta_Q^{'}\right.\right)\log P\left(w\left|\theta_D\right.\right) + \eta \log P\left(t\left|\theta_D\right.\right) \tag{4}$$

where $\eta > 0$ is the weight for the term t. In our experiments, α was set to 0.2 and η was set to 0.05 which is in the same range as the weights of the top 10 terms of the feedback model θ_F.

Let $MAP\left(Q \cup S\right)$ be the retrieval performance when S is the expansion set and $MAP\left(Q \cup S \cup t\right)$ be the retrieval performance after adding t to S with weight η. We can now define the relative marginal gain in retrieval performance as follows:

$$RMG\left(S, t\right) = \frac{MAP\left(Q \cup S \cup t\right) - MAP\left(Q \cup S\right)}{MAP\left(Q \cup S\right)} \tag{5}$$

Using relative marginal gain, we labeled t as follows: good if $RMG\left(S, t\right) \geq \delta$, bad if $RMG\left(S, t\right) \leq -\delta$ and neutral otherwise. Here $\delta > 0$ is a cutoff which in our experiments was set to 0.005.

Testing for Inconsistency. Let T_k denote the set of top k terms of the feedback model θ_F and t be a term[3]. For each of the expansion sets $T_1, \ldots, T_{25}$, we assigned a label $l_k \in \{$good, bad neutral$\}$ to t depending on the effect of adding t to T_k. We call t consistent if all the labels l_k, $k = 1, \ldots, 25$ are identical and inconsistent otherwise. A consistent term is one which behaves the same way with each of the expansion sets $T_1, \ldots, T_{25}$. Using this procedure we estimated N_Q, the number of inconsistent terms in each topic.

Test Results. We used several collections in our study: CLEF 2000-02, CLEF 2003,05,06 , AP (Associated Press 88-89, TREC Disks 1 and 2), WSJ (Wall Street Journal , TREC Disks 1&2), SJM (San Jose Mercury, TREC Disk 3) and TREC Disks 4&5 (minus the Congressional Record). Table 1 shows the mean and standard deviation of the number of inconsistent terms in the topics for several test collections and Figure 2 shows the histogram of inconsistent terms for the AP collection. We observed that very few topics have a small number of inconsistent terms. For most topics in the collections, 30-60 terms out of the top 100 are inconsistent and about 50% of the terms are inconsistent on an average.

[3] Model-based feedback produces a large number of feedback terms. We used only the top 100 terms.

Table 1. Inconsistent Terms (with top 10 documents as feedback)

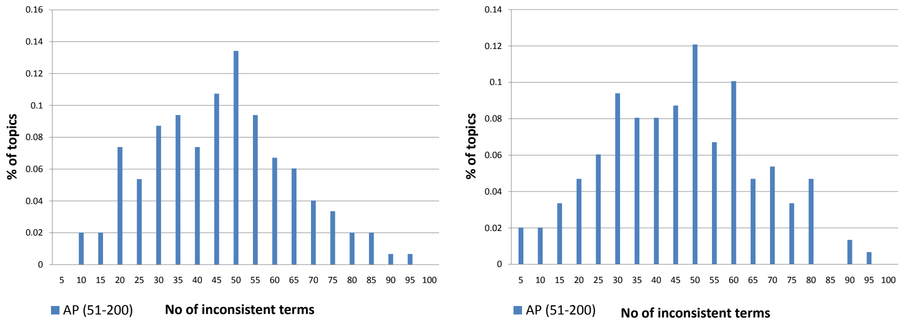
Collection	Mean	Std.Dev.
CLEF(1-140)	58.13	21.24
CLEF(141-200,251-300)	49.61	21.06
AP(51-200)	44.37	18.13
WSJ(51-200)	50.69	18.46
SJM(51-150)	51.46	20.43
Disk 4&5(301-450)	48.62	18.41

Table 2. Inconsistent Terms (with relevant documents as feedback)

Collection	Mean	Std.Dev.
CLEF(1-140)	52.42	25.22
CLEF(141-200,251-300)	49.35	23.44
AP(51-200)	44.21	19.66
WSJ(51-200)	50.62	18.90
SJM(51-150)	51.28	21.83
Disk 4&5(301-450)	47.06	18.83

These statistics unequivocally tell that inconsistency is a major problem and can not be ignored while selecting terms.

In a different experiment, we used the relevant documents of a topic to estimate its topic model. This was to verify whether the source of the inconsistent terms in PRF was non-relevant documents in the feedback. We repeated the inconsistency test with the top 100 terms of the topic model estimated directly from the relevant documents. Table 2 shows the mean and standard deviation of the number of inconsistent terms in the topics for several test collections. We observed that the mean and standard deviation were similar to those in the previous experiment. This means that even if relevant documents are provided as feedback, inconsistency remains an important issue.



(a) Using top 10 feedback documents

(b) Using relevant documents

Fig. 2. Distribution of N_Q in the AP collection

4 Finding Good Expansion Set by Spectral Partitioning

We now describe a principled approach for finding good expansion sets that takes into account term interactions. The key idea here is to form a weighted term-document matrix and partition it into two sets using Singular-Value Decomposition (SVD) [7]. Geometrically, the principal singular vector of the term-document matrix gives the direction that captures most of the spread (or variance) in the data. In pattern recognition and machine learning literature, this is also known as principal components analysis and is known to provide a very good low-dimensional representation of the data. Spectral partitioning techniques are very effective in practice and have been used successfully in many applications [6,13]. Further, they are highly useful in understanding global properties of a phenomenon using local interactions. Note that the covariance matrix obtained by multiplying the (centered) term-document matrix with its transpose captures pair-wise interactions. Higher powers of the covariance matrix in turn capture higher order interactions.

4.1 Partitioning Algorithm

Let A be a matrix whose rows represent the candidate feedback terms[4] $\{t_i\}_{i=1}^{m}$ and the columns represent the feedback documents $\{D_j\}_{i=1}^{n}$. Let $[A]_{ij} = a_{ij}$ be a measure of the interaction between term t_i and document D_j. We express a_{ij} as $a_{ij} = \text{global}\,(t_i) * \text{local}\,(t_i, D_j)$ where $\text{global}\,(t_i)$ is a global weighting function and $\text{local}\,(t_i, D_j)$ is a local weighting function.

The global weighting function measures the informativeness of terms with respect to the feedback documents. Our choice for this function is the following:

$$\text{global}\,(t) = \ln\left(\frac{n_t}{n}\Big/\frac{df_t}{N}\right) \tag{6}$$

where n (and resp. N) is the number of feedback documents (and resp. number of documents in the collection) and n_t (and resp. df_t) is the document frequency of t in the feedback corpus (and resp. document frequency of t in the collection). We can write

$$\text{global}\,(t) = \ln\frac{N}{df_t} - \ln\frac{n}{n_t} = idf_t - idf'_t \tag{7}$$

where $idf'_t = \ln\frac{n}{n_t}$ is the idf of t in the feedback corpus. It can be easily shown that $\text{global}\,(t) \le \ln\frac{N}{n}$ with equality holding only when $n_t = df_t$. Thus, according to the global weighting function, a term t is more informative than another term s if $idf_t - idf'_t > idf_s - idf'_s$ or equivalently if $idf_t - idf_s > idf'_t - idf'_s$. Therefore, from the point of view of PRF, t can be more informative than s even when $idf_t < idf_s$. Finally, our choice for $\text{local}\,(t_i, D_j)$ is $P\,(t|D)$, the smoothed unigram probability of the term t in document D [14].

We center the matrix A such that the mean of the row vectors is the $\vec{0}$ vector. The Singular Value Decomposition (SVD) of the term-document matrix A is then the following:

$$A = U\Sigma V^T \tag{8}$$

where U and V are orthogonal matrices and Σ is a diagonal matrix.

[4] Candidate expansion terms are those terms from the feedback documents whose $idf > \ln 10$ and collection frequency ≥ 5. When there are more than 100 such terms we take the top 100 according to their frequency in the feedback documents.

The sign of the terms in the principal left singular vector $\vec{u_1}$ suggests a principled way to partition the terms into two sets [7]. We form the first set, S^+ by taking those terms whose sign is positive in the principal singular vector. Similarly, we form the second set, S^- by taking those terms whose sign is negative. We remove terms from S^+ and S^- that have an absolute weight below a threshold.

4.2 From Partitions to Feedback Model

There are several ways in which we can form a feedback model using S^+ and S^-. In our experiments we used two very simple methods as our goal was to mainly validate the goodness of the expansion sets produced by spectral partitioning. In the first method (SU), we assigned uniform probability to all the terms in the set and in the second (SF), we assigned a probability proportional to the frequency of the term in the feedback documents. In both methods, we formed feedback models θ_F^+ (which allocates non-zero probability to only terms of S^+) and θ_F^- (which allocates non-zero probability to only terms of S^-). The two models θ_F^+ and θ_F^- represent two different choices for expansion. As our goal in this study was to demonstrate that spectral partitioning separates the good set of terms from the bad, we used the one that gave the best results for the topic.

5 Experimental Results

We tested our spectral partitioning idea on the following test collections: CLEF 2000-02, CLEF 2003,05,06 , AP (Associated Press 88-89, TREC Disks 1 and 2), WSJ (Wall Street Journal , TREC Disks 1&2), SJM (San Jose Mercury, TREC Disk 3) and TREC Disks 4&5 (minus the Congressional Record). We used two baselines, language model (LM) with two stage Dirichlet smoothing and the mixture feedback model (MF) [4]. We interpolated the feedback model with a weight of 0.5. We stemmed the words using the well-known Porter stemmer and removed stop-words from topics and documents.

Table 3 shows the retrieval results for various models. We did t-test to determine the significance of the results. Both SF and SU gave substantially better results than the LM and MF baselines on all test collections. We observed $> 15\%$ improvement in MAP over LM for all the test collections. The spectral methods had substantially better P@10 compared to both LM and MF. This means that spectral expansion was able to retrieve relatively larger number of relevant results in the top 10. Further, we observed improvement in all performance metrics. Finally, the performance of SU was comparable with that of SF which means that the expansion sets are robust to perturbations in weights.

Table 4 shows the expansion sets for three topics. Firstly, we observe that the terms in each set are topically coherent. Consider Topic 311 of TREC for instance. All the expansion terms found by our Spectral Partitioning algorithm are topically related to the query **Industrial Espionage**. In Topic 112 of CLEF, we observe that the expansion terms found by our Spectral Partitioning algorithm

Table 3. Retrieval results (** = significant at $p < 0.01$ over LM) (ˆˆ = significant at $p < 0.01$ over MF)

Collection	Model	P@10	P@100	MAP	% Improvement	Recall
CLEF (1 - 140)	LM	0.3828	0.1181	0.4338	-	0.8936
	MF	0.4148	0.1310	0.4417	1.82%	0.9348
	SU	0.4459	0.1383	0.5037	16.11% **ˆˆ	0.9578
	SF	0.4484	0.1398	0.503	15.95% **ˆˆ	0.9632
CLEF (141 - 200, 251 - 300)	LM	0.3684	0.1573	0.3808	-	0.8172
	MF	0.3836	0.1694	0.4053	6.43% **	0.9239
	SU	0.4296	0.1737	0.4516	18.59% **ˆˆ	0.9357
	SF	0.4362	0.1739	0.4474	17.49% **ˆˆ	0.9345
AP (51 - 200)	LM	0.4450	0.2655	0.2772	-	0.6504
	MF	0.4732	0.3026	0.327	17.97% **	0.7217
	SU	0.5201	0.3303	0.3641	31.35% **ˆˆ	0.7565
	SF	0.5315	0.3312	0.3648	31.60% **ˆˆ	0.7564
WSJ (51 - 200)	LM	0.4573	0.2611	0.2660	-	0.6438
	MF	0.4773	0.2884	0.3027	13.79%**	0.6971
	SU	0.5193	0.2975	0.3238	21.73% **ˆˆ	0.7106
	SF	0.5113	0.2969	0.3213	20.79% **ˆˆ	0.7173
SJM (51 - 150)	LM	0.3043	0.1572	0.2074	-	0.6173
	MF	0.3234	0.1736	0.2350	13.31%**	0.6773
	SU	0.3649	0.1832	0.2601	25.41% **ˆˆ	0.6916
	SF	0.3691	0.1816	0.263	26.81% **ˆˆ	0.6992
Disks 4& 5 (301 - 450)	LM	0.4247	0.1987	0.2275	-	0.5359
	MF	0.4360	0.2152	0.2505	10.11% **	0.5746
	SU	0.4693	0.2385	0.2848	25.19% **ˆˆ	0.6153
	SF	0.4707	0.2413	0.2876	26.42% **ˆˆ	0.6205

Table 4. Expansion set using Spectral Partitioning

Pulp fiction		**Machine Translation**		**Industrial Espionage**	
(Topic 112, CLEF)		(Topic 63, TREC 3)		(Topic 311, TREC 6)	
Term	$P(t\|\theta_F)$	Term	$P(t\|\theta_F)$	Term	$P(t\|\theta_F)$
movi	0.25	comput	0.28	vw	0.23
film	0.20	english	0.21	gm	0.17
travolta	0.20	word	0.14	german	0.14
tarantino	0.18	languag	0.09	lopez	0.13
actor	0.09	human	0.07	investig	0.13
quentin	0.06	recogn	0.06	opel	0.07
cann	0.02	voic	0.05	motor	0.07
		pen	0.05	volkswagen	0.05
		dictionari	0.04	wolfsburg	0.03

include the names of the director and lead actor of the movie **Pulp Fiction**. Finally, in Topic 63 of AP too, the selected expansion terms are topically related to **Machine Translation**.

6 Conclusion

We showed that the term independence assumption for the selection of good expansion terms does not hold in practice through a thorough study of the effect of an expansion term in the presence of other terms. In practice, about 50% of the expansion terms are inconsistent, i.e. they behave differently with different expansion sets. Our empirical finding implies that good expansion sets can not be discovered by methods that make independence assumption in general. As a principled method of discovering good expansion sets, we proposed spectral partitioning of term-term interaction matrix which takes into account term interactions of all orders. We demonstrated that the expansion sets produced by spectral partitioning give substantially better retrieval results than both language model and model-based feedback on topics from several test collections.

In a future study, we will explore more sophisticated methods for forming the feedback model from the partitions produced by our method. For instance, we can leverage weights of terms in the principal left singular vector while forming the feedback model. Another direction of research is how to choose between θ_F^+ and θ_F^- for a given topic.

References

1. Buckley, C., Salton, G., Allan, J.: Automatic retrieval with locality information using smart. In: TREC, pp. 59–72 (1992)
2. Carpineto, C., Romano, G.: Towards more effective techniques for automatic query expansion. In: Abiteboul, S., Vercoustre, A.-M. (eds.) ECDL 1999. LNCS, vol. 1696, pp. 126–141. Springer, Heidelberg (1999)
3. Cao, G., Nie, J.Y., Gao, J., Robertson, S.: Selecting good expansion terms for pseudo-relevance feedback. In: SIGIR 2008: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 243–250. ACM Press, New York (2008)
4. Zhai, C., Lafferty, J.: Model-based feedback in the language modeling approach to information retrieval. In: Proceedings of Tenth International Conference on Information and Knowledge Management, pp. 403–410 (2001)
5. Lavrenko, V., Croft, B.W.: Relevance based language models. In: SIGIR 2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 120–127. ACM Press, New York (2001)
6. Chung, F.R.K.: Spectral Graph Theory (CBMS Regional Conference Series in Mathematics), February 1997. Cbms Regional Conference Series in Mathematics, vol. 92. American Mathematical Society, Providence (1997)
7. Meyer, C., Basabe, I., Langville, A.: Clustering with the svd. In: Workshop on Numerical Linear Algebra, the Internet and its Applications, Monopoli (2007)

8. von Luxburg, U.: A tutorial on spectral clustering. Technical Report 149, Max Planck Institute for Biological Cybernetics (August 2006)
9. Efthimiadis, E.N.: Query expansion. Annual Review of Information Systems and Technology 31, 121–187 (1996)
10. Zhai, C.: Statistical language models for information retrieval a critical review. Found. Trends Inf. Retr. 2(3), 137–213 (2008)
11. Kullback, S., Leibler, R.A.: On information and sufficiency. The Annals of Mathematical Statistics 22(1), 79–86 (1951)
12. Tao, T., Zhai, C.: Regularized estimation of mixture models for robust pseudo-relevance feedback. In: SIGIR 2006: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 162–169. ACM, New York (2006)
13. Spielman, D.A., Teng, S.: Spectral partitioning works: Planar graphs and finite element meshes. Technical report, Berkeley, CA, USA (1996)
14. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst. 22(2), 179–214 (2004)
15. Lafferty, J., Zhai, C.: Document language models, query models, and risk minimization for information retrieval. In: SIGIR 2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 111–119. ACM, New York (2001)
16. Robertson, S.E.: On term selection for query expansion. J. Doc. 46(4), 359–364 (1990)
17. Smeaton, A.F., van Rijsbergen, C.J.: The retrieval effects of query expansion on a feedback document retrieval system. Comput. J. 26(3), 239–246 (1983)
18. Robertson, S.E., Jones, S.K.: Relevance weighting of search terms. Journal of the American Society for Information Science 27(3), 129–146 (1976)

An Effective Approach to Verbose Queries Using a Limited Dependencies Language Model

Eduard Hoenkamp[1], Peter Bruza[2], Dawei Song[3], and Qiang Huang[3]

[1] University of Maastricht
hoenkamp@acm.org
[2] Queensland University of Technology
p.bruza@qut.edu.au
[3] Robert Gordon University
{d.song,q.huang}@rgu.ac.uk

Abstract. Intuitively, any 'bag of words' approach in IR should benefit from taking term dependencies into account. Unfortunately, for years the results of exploiting such dependencies have been mixed or inconclusive. To improve the situation, this paper shows how the natural language properties of the target documents can be used to transform and enrich the term dependencies to more useful statistics. This is done in three steps. The term co-occurrence statistics of queries and documents are each represented by a Markov chain. The paper proves that such a chain is ergodic, and therefore its asymptotic behavior is unique, stationary, and independent of the initial state. Next, the stationary distribution is taken to model queries and documents, rather than their initial distributions. Finally, ranking is achieved following the customary language modeling paradigm. The main contribution of this paper is to argue why the asymptotic behavior of the document model is a better representation then just the document's initial distribution. A secondary contribution is to investigate the practical application of this representation in case the queries become increasingly verbose. In the experiments (based on Lemur's search engine substrate) the default query model was replaced by the stable distribution of the query. Just modeling the query this way already resulted in significant improvements over a standard language model baseline. The results were on a par or better than more sophisticated algorithms that use fine-tuned parameters or extensive training. Moreover, the more verbose the query, the more effective the approach seems to become.

1 Introduction

Imagine (or perhaps recall) that you just came back from a well-deserved vacation in the South Pacific. When someone asks you about your vacation, you are happy to recount how it was. First you tell it to the people at home, then to your neighbors, then to your colleagues at work. At first there will be much variation in your story, but by and by all has been said, and the rendition of your experience becomes stable, only mentioning the essential parts. Or think of

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 116–127, 2009.

an event that lands as late breaking news on your paper's front page. As days go by, the story may reappear a few times, but eventually all has been said.

Now suppose a search engine would need to return the most relevant (as opposed to the most entertaining) story about your vacation. Should it be one from the earlier stages where it still meandered haphazardly along all that happened? Or one of the later more concise and orderly accounts?

Let us look at this phenomenon from the language modeling perspective to IR [1]. In this paradigm a text is viewed as a sample from a stochastic source that produces words according to some distribution. With the vacation story, you were the source, and your stories were different samples from that source. As the source is assumed to be stochastic, the words and their frequencies will change from one account to the next, as in the case of your stories.

Without a model of the underlying process, however, it would be difficult to reconstruct the distribution of the source from the samples alone. Therefore, language models can be distinguished by how they model the source and by how the distribution is derived from the samples. As current language models don't use an explicit representation of the meaning of documents, we can illustrate our approach with a simple abstract example. Assume a language of just the words a and b, and two documents $D_1 = [a\ a\ a\ a\ a\ b\ b\ b\ b\ b\ a]$ and $D_2 = [a\ b\ a\ b\ a\ b\ a\ b\ a\ b\ a\ b]$. Using $Q = [a\ b\ a\ b\ ]$ as the query (or topic), which document would be considered the most relevant for a given language model? In the multi-bernoulli model [1], D_1 and D_2 would get the same score, as all words in the query are also in the documents. The multinomial unigram model [2] also assigns the same score because the frequencies of a and b are the same in D_1 and D_2 and hence the $p(Q|D) = \prod_i p(q_i|D)$ are the same. If Q were extended with a word c that does not appear in the documents, so that smoothing [3] was called for, words would be discounted by the same amount, and again the documents would receive the same score. Basically, we are trying to estimate a relevance model (1) without further knowledge about the corpus, (2) under the assumption that the term occurrences are independent, and (3) in the absence of training data. These issues have received much attention lately. For example, several researchers have studied bigrams and trigrams [2] or even studied the optimal distance over which to consider dependencies in general [4,5] or based on natural language constraints [6]. Metzler and Croft [5] in particular distinguished among full independence, sequential dependence, and full dependence. The terms mean what they suggest: in sequential dependence the ranking of a document depends only on the dependency of adjacent words, whereas in full dependence any clique of words is to be considered. In this paper we consider a fourth option, halfway between sequential and full dependence, namely when a word comes after another, but separated by words in between. For example, in D_1 and D_2 above, one can accumulate the distances from every a to every b to derive a probability that a is followed by b. In the example, this probability is much lower for D_1 than for D_2. Imagine that, as in the vacation story that was told over and over again, the sources of D_1 and D_2 would go on for a long time producing one new document after another according to their distributions. If we assume for concreteness a dependency of no more than five words, then (as we will see) in the long run a would appear about as often as b for D_2 but twice as often for D_1. This is obviously different from the word counts that would suggest a 50% probability for each. Moreover, the distribution in the long run seems to reflect the impression that D_2 is more like Q than is D_1. This paper will show how the term dependencies of a particular document predict the asymptotic behavior of its source, and with it the term distribution that would be observed if the source would continue to produce new documents.

The sections that follow show how the approach of asymptotic behavior relates to other language models, and how it accomplishes the following objectives:

– It shows that under very realistic, plausible, and elementary conditions the *source underlying a document is ergodic*, and therefore a stationary distribution to represent the source can be derived from just one document,
– It shows how documents can be ranked based on their underlying stationary distributions,
– It shows how an initial (ad hoc) distribution for a document can be established, based on a semantic approach called the *Hyperspace Analog to Language* (HAL).

2 The Document Source as an Ergodic Chain

One reason that language models use lower order dependencies is the (in)tractability of the Bayesian chain rule. Another is often simply a lack of knowledge about higher order dependencies. Yet, in practice, bigrams already give a reasonable improvement over unigrams [7]. In addition, [2] and others have shown that an interpolation of unigram and bigram models performs well.

The practical considerations aside, the question remains whether higher order dependencies would lead to better models, even if it is tempting to assume the affirmative. To begin answering the question, it is important to realize that the current approach to language modeling is applicable to any stochastic source and the languages they produce (human, machine, or perhaps of unknown origin). The models pay no heed to the fact that the documents to be modeled are produced by humans. Yet this throws out particular constraints that could make the methods more tractable. Some constraints can be borrowed from cognitive science, some follow directly from confining the languages under consideration to natural language:

– Many cognitive phenomena can be understood sufficiently well in terms of word-pairs. Pertinent examples can be found e.g. in the research on memory [4], work as mentioned above on the ‘semantic space’ [8], and results from old theories on ’spreading activation’ [9] to recent brain (ERP) studies [10]. This supports the view that the source underlying the document can be modeled as a (first order) Markov process.
– Words in a natural language corpus can be separated by any number of intermediate words. (Think of adding an extra adjective before a noun.) This

means there cannot be any cycles in the process. Identifying words with the states of the process then means that the Markov chain is *aperiodic.*
- You can always get from one word to another by continuing to produce text (words can never be used up). Consequently, the Markov chain is *irreducible.*

The first point was already proposed by Shannon in his famous article [11], without the backup from cognitive science. The next two points, that the Markov process is both aperiodic and irreducible means that it is *ergodic.* An ergodic chain has the property that in the long run it reaches a stationary distribution (also called stationary kernel, or steady state), irrespective of the initial state. It is easy to sample a document and generate a new one on the basis of its distribution; see the examples in [11], or any of the many sites on the web that offer programs to do this[1]. What we would like to compute however is the distribution of the source underlying the document. Or in the metaphor of the introduction, we would like to model the final stable and concise story as the most relevant to the query about the vacation. With little knowledge of the source, one could use a Gibbs sampler, i.e. generate a long series of documents and sample until the distribution seems to converge. The Gibbs sampler was proposed for example by Wei and Croft [12] to estimate the joint distribution of their LDA model. Besides the benefits of that model, there are several issues to overcome: (1) it is computationally demanding, (2) it is hard to know when the process has converged, and (3) The fixed point may not be unique and e.g. depend on the initial state (the Gibbs sampler *assumes* the process is ergodic, but LDA does not imply this). The derivation above that the process we advance here is indeed ergodic obviates all three issues at once: The stationary distribution of the Markov chain can be efficiently computed (as we will show in the next section), no continued sampling is required to know whether the distribution has converged, and it is guaranteed to be unique.

Note, first, that the properties mentioned to derive this result are valid for natural languages in general. This means that the method may be used for languages other than English (and which are increasingly visible on the Web). Second, it also answers the question about the higher order dependencies, in that it is unlikely that these will contribute much to improving search results. With the answer comes an other question to the fore: how to compute the lower order dependencies given the documents. The next section offers a proposal, one we will use in an experiment further on, but it is by no means meant as the last word on finding initial distributions.

3 Deriving the Initial Distribution

In language modeling, the document source represents the author producing the document. As an author could produce different renderings of the same

[1] For example http://www.nightgarden.com/infosci.htm explains the procedure and links to a 'Shannonizer' where you can input text, or refer to a URL, to generate a text based on bi-grams.

story, these renderings would be different samples of the source, and so the term distribution could differ from one document to the next.

Fortunately, the ergodic chain has a property that is very useful here, namely that its asymptotic behavior is independent of the initial state. In other words, if one would continue to sample the source, then in the long run it would not matter what sample, i.e. what document, was observed first; the asymptotic behavior would be the same. What remains then, is to derive an initial distribution given the document.

Box 1

Given an n-word vocabulary, the HAL space is represented as a $n * n$ matrix constructed by moving a window of size w over the corpus ignoring punctuation, sentence, and paragraph boundaries. The strength of co-occurence decreases with the number of intervening words. Instead of an large-scale corpus, let us take just the sentence *The effects of spreading pollution on the population of Atlantic salmon.*

	the	effects	of	spreading	pollution	on	population	atlantic	salmon
the		1	2	3	4	5			
effects	5								
of	8	5		1	2	3	5		
spreading	3	4	5						
pollution	2	3	4	5					
on	1	2	3	4	5				
population	5		1	2	3	4			
atlantic	3		5		1	2	4		
salmon	2		4			1	3	5	

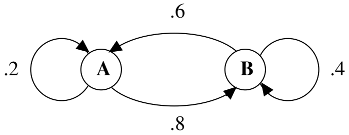
The table above shows the HAL matrix for a window size of 5. Take e.g. the entry for 'population'. To find the distance to 'pollution', go backward starting at 'population' with strength 5 (for 'the') counting down to 3 for 'pollution'.

This is where language models differ greatly from one another. As we mentioned in the introduction, an important distinction lies in the degree of term dependency that is assumed. In this paper we follow the approach of Lund and Burgess [13] who computed co-occurrence statistics from a rich source of spontaneous conversations: Usenet newsgroups. They called the representation of these statistics the 'Hyperspace Analog to Language' or HAL. HAL is computed by sliding a window over the corpus and assigning weights to word pairs, inversely to the distance from each word to every other in the window. This results in a word by word matrix with the accumulated word distances in the cells. **Box 1** may further clarify how the HAL matrix is computed. Lund and Burgess [13] experimented with various window sizes, which obviously produce different HALs. They found that the associations that people make between word-pairs can best be modeled with a window size between 8 and 10. Other experiments confirmed that size as optimal to describe the correlation between word co-occurrance in corpora and strength of word association [8]. (The window size of 5 in **box 1** was chosen for clarity of exposition, not to model people's word associations.)

Box 2

For readers unfamiliar with the Markov approach, the essential steps in the algorithm are illustrated below. Assume a language of just the words a and b, with dependencies as defined by the transition probabilities in matrix H. H defines a Markov chain, where state **A** ouputs a and state **B** outputs b.

$$\begin{array}{c|cc} H & a & b \\ \hline a & .2 & .6 \\ b & .8 & .4 \end{array} \quad \Rightarrow$$



For initial state s_0 (e.g. **A** if started with word a), the next state is given by $s_1 = s_0 * H$, where

$$H = \begin{pmatrix} .2 & .6 \\ .8 & .4 \end{pmatrix}$$

followed by $s_2 = s_1 * H = s_0 * H^2, ..., s_n = s_0 * H^n$ with

$$H^n = \tfrac{1}{.8+.6} \begin{pmatrix} .6 & .6 \\ .8 & .8 \end{pmatrix} + \tfrac{-0.4^n}{.8+.6} \begin{pmatrix} .8 & -.6 \\ -.8 & .6 \end{pmatrix}$$

which converges to: $\lim_{n\to\infty} H^n = \begin{pmatrix} .4286 & .4286 \\ .5714 & .5714 \end{pmatrix}$.

so the Markov chain becomes stationary with $P(a) = .4286$ and $P(b) = .5714$, independent of the initial state. (The formal derivation was only given to show the convergence. The stationary distribution can also be computed directly from the transition matrix.) In the same way these values can be obtained for the examples in the introduction. Computing the HAL matrix with window of size 4, the distributions converge to:
$D_1 = [a\ a\ a\ a\ a\ b\ b\ b\ b\ b\ b\ a]$, $P(a) = .36$ and $P(b) = .64$
$D_2 = [a\ b\ a\ b\ a\ b\ a\ b\ a\ b\ a\ b]$, $P(a) = .49$ and $P(b) = .51$
$Q = [a\ b\ a\ b\ ]$, $P(a) = .44$ and $P(b) = .56$
Computing the Kullback-Leibler divergence yields
$KL(Q||D_1) = .017$, and $KL(Q||D_2) = .007$, so D_1 diverges more from Q than D_2, and therefore D_2 is ranked as more relevant.

If a word is connected to a second word via a small number, than it is more likely followed by that word than if the number had been high (e.g. the table shows that 'of' is more likely to be followed by 'the' than the other way around). Based on this observation, the HAL matrix is transformed into a transition probability matrix *pHAL* by normalizing the row vectors (see e.g. [14]). So, to find the document source distribution for a document requires only two steps:

1. Compute the ad-hoc distribution, in our case pHAL,
2. Compute the stable distribution (epi-HAL).

This *epi-HAL*, for 'ergodic process interpretation of HAL', is easy to compute in several ways, which follow from the ergodic property[2]. Doing this for all documents produces a source representation for each document. The same can be done for the query, which would represent the searcher. To rank the documents in order of relevance to the searcher, the documents are not compared to the query directly (as in the vector space model) but the sources are compared. Researchers in the language modeling community use the Kullback-Leibler (KL) divergence to compare distributions, and so will we. The algorithm is explained in **Box 2** using a very simple language for clarity.

The main goal of this paper is to explain and more formally justify our approach, which is what we did in the sections so far. Note that a longer query corresponds to a larger sample from the source, so one would expect that longer queries would automatically be more effective. In light of an observation recently published by Bendersky and Croft [15], this needs empirical verification. Therefore, the next section will add a more practical justification by showing that even a straightforward and simple implementation of our approach can already compete with a closely related but much more sophisticated language model.

4 Implementation and Evaluation

There certainly are other language models that use a Markov approach. Besides [12] mentioned earlier, notably Cao, Nie, and Bai [16] use the Markov chain for a similar reason as we do, namely to find a stable distribution to represent the document. But there are a number of choices made in [16] that we do not depend on: we do not use WordNet (for semantic relationships), there are several parameters we do not have to set, and we don't use training for optimization. Furthermore, although the authors of [16] make use of a stationary distribution, there are several issues with their approach: (1) it is computationally demanding, (2) it is hard to know when the process has converged, and (3) there is no indication, let alone a proof, that the algorithm has only one fixed point. So, e.g. depending on the initial state, their stationary distribution may or may not be

[2] First, computing a HAL matrix is approximately n^2 in the length of the text, but since it is additive, we distribute it over a grid for efficiency. Second, the stationary distribution of the Markov chain can quickly be computed without sampling (it is the eigenvector with eigenvalue 1).

the one sought after. The observation above that the process we advance is ergodic, obviates all three issues at once: The final distribution of the Markov chain can easily be computed without sampling (it is the eigenvector with eigenvalue 1), it converges very fast, and it is guaranteed to be unique.

We will now turn to an experimental evaluation of our ergodic process interpretation of HAL (epi-HAL). The experiment is comparable to that reported for the relevance model of Lavrenko & Croft [17], following a pseudo-relevance feedback paradigm. We first compute a document ranking in response to a query Q. The top n documents are used to derive a distribution M_{epi}^{n} by computing the epi-HAL over this collection. Similarly, M_{epi}^{Q} is computed for the query. These are used in turn to define a mixture model (cf. equation (15) in [17]).

$$\Pr(w|Q) = \lambda \Pr(w|M_{\text{epi}}^{Q}) + (1 - \lambda) \Pr(w|M_{\text{epi}}^{n})$$

The documents are re-ranked using the KL-divergence, and we use the standard baseline unigram LM in the Lemur toolkit. We set the number of feedback documents, n, to 30. For query extension we used 300 terms. Others use different values here, and such differences are to be expected as the distributions are calculated differently, and there is no better way known than to establish these numbers empirically. With these numbers (or another choice) the query model $M_Q = \Pr(w|Q)$ can be computed. Subsequently, documents are re-ranked via $KL(M_Q||M_D)$, where M_D corresponds to a document language model. In our case, M_D is delivered by the baseline language model.

We noted earlier that we expect our approach to work better with longer queries, because a longer query means a larger, and hence more representative, sample from the source. (Note that one could see pseudo-relevance feedback as an attempt to make the query longer.) Such longer, or more verbose, queries also seem more representative of the way humans communicate their information needs, compared to typing in a few query words. Bendersky and Croft in their recent paper [15] simulate increasing verbosity by using TREC topics and take the *description* field as a more verbose version of the *title* field. If our intuition (and theirs) were correct one would expect better results for the description than for the title. They found, however, precision to go down substantially for the description. Bendersky and Croft's intuition is that the focus on the key concepts gets blurred as it were by the verbosity surrounding it. We think this intuition leads to two questions, or rather, predictions:

- Assuming the explanation is valid, what would this predict if the description and title were taken together as the new query? Such a query could become less effective then the description, because it is more verbose. Alternatively, it could become more effective because someway the key concept becomes more prominent. Or, combining the two arguments, a safer guess might be that it lands between the efficacy of description and title in isolation. So this has to be investigated empirically.
- Given that the HAL representation captures the semantic relationships between words in the corpus [8,13], the cohesion between key concepts would

> be enhanced by the co-occurance of words expressing the concepts. In turn, that would increase the weight of certain words by increasing their value in the joint probability distribution (the query model). And so it would predict a higher effectiveness of title and description together, than either in isolation. (Note that Bendersky and Croft propose to enhance the focus on the key concept using a learning algorithm to weight the words in the query. A different approach that might lead to the same result.)

We will see how these predictions fare for various combinations of title and description.

4.1 Experimental Results

Besides the title and description from the TREC topics, we also added the narrative, as it is even more verbose than the description. We shall first present the results for the now classical AP corpus, and present some initial results with the ROBUST04 collection that Bendersky and Croft used. The results of AP8889 are in Table 1. We used topics 101-150 of AP8889 because it has an exclusion clause in the narrative. For example topic 102, describing Laser research for SDI, ends with "However, a document clearly focused on use of low-power lasers in consumer products, surgical instruments, or industrial cutting tools is NOT relevant." We used two versions of the narrative, one with, and one without the exclusionary clauses. This way we could get an indication of the effect of verbosity: with the exclusion clause intact, the query is obviously more verbose, but more off focus. We used the Lemur search engine toolkit for the computations. The following models were used: the baseline language model provided by Lemur, the relevance model proposed by Lavrenko and Croft [17], and the stable distribution approach we advance in the current paper. The results for the stable distribution was also computed in Lemur, using its smoothing model, but taking the stable distribution as query model. For the AP corpus and the given topics, the precision goes up with increasing verbosity. The baseline precisions breaks down going from title only to description only, as was observed previously by Bendersky and Croft. Both the relevance model and the

Table 1. Comparing precision for various degrees of verbosity and different language models for AP8889 topics 101-150. *title*, *desc*, and *narr* stand for the corresponding TREC fields. $narr_{-rc}$ stands for narratives with the topic 101-150 exclusion clauses removed. 'Baseline' is from Lemur's default simple language model, 'Relevance model' follows [8], and 'epi-HAL' is the model proposed in the current paper.

Topics 101-150		$< title >$	$< desc >$	$< title, desc >$	$< title, narr >$	$< title, narr_{-rc} >$
Baseline	MAP	23.6	22.7	28.8	31.7	31.9
	prec@5	41.2	44.4	48.8	50.8	50.0
Relevance model	MAP	29.5	29.0	32.3	32.8	33.0
	prec@5	43.6	44.0	42.8	48.8	46.4
Stable Distribution (epi-HAL)	MAP	32.3	32.4	35.7	39.5	39.3
	prec@5	46.0	46.4	46.2	60.0	58.2

Table 2. ROBUST04 results, comparing mean average precision (MAP) for title, description, and their combination, for baseline and epi-HAL. Number of documents: 528,155, topics 301-450 and 601-700.

ROBUST04	$< title >$	$< desc >$	$< title, desc >$
Baseline	25.7	24.8	28.7
epi-HAL	31.1	31.0	33.1
Bendersky and Croft	25.28	26.2	-

epi-HAL model appear to be less sensitive to this break down. And as both are feedback models, perhaps it is the feedback that dampens the effect. For every model, however, when title and description are combined, the precision rebounds completely, and surpasses the precision over either in separation. So verbosity cannot be the sole ground for the lack of precision of the description by itself. Table 2 offers a preliminary comparison of epi-HAL with the best performing published results of a state-of-the-art model by Bendersky and Croft [15]. This variation adopts a machine learning approach to identify which noun phrases in the description are key and use the key concepts to boost retrieval of verbose queries. No results were reported for this model on both title and description as Bendersky and Croft did not run the model on the combination of both. The MAP of 26.2 reported for are those for the KeyConcept[2]<desc> variation of the model.

The results point in the same direction as the AP experiment: the baseline shows the precision collapse for description only, the feedback dampens the effect, precision recovers when title and description are combined, and for our approach the precision increases with verbosity. The epi-HAL largely outperformed the baseline by 21%, 25% and 15% respectively on the use of titles, descriptions, and titles plus descriptions, and outperformed the Bendersky and Croft model by 21% and 18% on the use of titles and descriptions respectively.

Note that these data are still preliminary for a detailed and more conclusive comparison with the learning approach of Bendersky and Croft, and are only cautiously indicative. However, as the descriptions of the query topics of the ROBUST04 collection are more verbose and grammatically complex than those of the W10g and GOV2 collections, we put forward the hypothesis that the encouraging performance of the epi-HAL model is due to the ergodic process having more description to process and hence stabilize to a more effective query representation. If so, this suggests performance improvements will be less pronounced on the W10g and GOV2 collections where the query topics are less verbose. Further experimentation is needed to bear this out.

5 Conclusions and Future Work

We derived a relatively simple language model, epi-HAL, that deviates in several respects from other language models proposed to date. Epi-HAL is based on the observation that texts are produced by humans. From this observation it follows that (1) there must be semantic dependencies underlying the documents, and (2) that the documents must obey surface constraints inherent to natural language. To represent the former, this paper derived the underlying semantics from the Hyperspace Analog to Language (HAL) a theory presuming that words that appear close together in text, will also be close in meaning. The surface constraints were represented by using an ergodic Markov chain.

We believe that current language models are overly general in that they do not incorporate these properties of natural language, the very fabric of the documents they purport to model. We compared a straightforward implementation of the proposed model with a sophisticated relevance model. Evaluation on TREC corpora showed that epi-HAL easily outperformed the relevance model for AP8889 and provided some initial encouraging results on the ROBUST04 collection. The epi-HAL model shows increased precision for more verbose queries, and therefore in the long run may respond more appropriately to the verbose inquiries humans typically engage in when communicating with one another.

The results of the experiments encourages us to pursue several avenues in future work. First, instead of modeling only the query by its stable distribution, the same can be done for the document model. Second a more elaborate and detailed experiment with larger corpora will be conducted. And finally, because the proposed model itself is relatively simple, its performance can be further improved via optimization of parameter settings as applied in current, much more sophisticated models.

References

1. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proceedings of the 21st Conference on Research and Development in Information Retrieval, pp. 275–281 (1998)
2. Song, F., Croft, W.B.: A general language model for information retrieval. In: Proceedings of the 22nd Conference on Research and Development in Information Retrieval, pp. 279–280 (1999)
3. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: SIGIR 2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 334–342. ACM Press, New York (2001)
4. Shiffrin, R.M., Steyvers, M.: The effectiveness of retrieval from memory. In: Oaksford, M., Chater, N. (eds.) Rational models of cognition, pp. 73–95. Oxford University Press, Oxford (1998)
5. Metzler, D., Croft, W.B.: A markov random field model for term dependencies. In: SIGIR 2005: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. ACM Press, New York (2005)
6. Gao, J., Nie, J.Y., Wu, G., Cao, G.: Dependence language model for information retrieval. In: Proceedings of the 27th annual international conference on Research and development in information retrieval, pp. 170–177. ACM Press, New York (2004)

7. Lafferty, J., Zhai, C.: Document language models, query models, and risk minimization for IR. In: Proceedings of the 24th Conference on Research and Development in Information Retrieval, pp. 111–119 (2001)
8. Burgess, C., Livesay, K., Lund, K.: Explorations in context space: Words, sentences, discourse. Discourse Processes 25, 211–257 (1998)
9. Anderson, J.: The Architecture of Cognition. Harvard University Press, Cambridge (1983)
10. Chwilla, D., Kolk, H.: Accessing world knowledge: Evidence from n400 and reaction time priming. Cognitive Brain Research 25, 589–606 (2005)
11. Shannon, C.E.: A mathematical theory of communication. Bell System Technical Journal 27, 379–423, 623–656 (1948)
12. Wei, X., Croft, W.B.: Lda-based document models for ad-hoc retrieval. In: SIGIR 2006: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 178–185. ACM Press, New York (2006)
13. Lund, K., Burgess, C.: Producing high-dimensional semantic spaces from lexical co-occurrence. Behavior Research Methods, Instruments, & Computers 28(2), 203–208 (1996)
14. Azzopardi, L., Girolami, M., Crowe, M.: Probabilistic hyperspace analogue to language. In: Proceedings of the 28th Annual ACM Conference on Research and Development in Infomration Retrieval (SIGIR 2005), pp. 575–576. ACM, New York (2005)
15. Bendersky, M., Croft, W.B.: Discovering key concepts in verbose queries. In: SIGIR 2008: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 491–498. ACM, New York (2008)
16. Cao, G., Nie, J.Y., Bai, J.: Using markov chains to exploit word relationships in information retrieval. In: The 8th Conference on Large-Scale Semantic Access to Content, RIAO 2007 (2007)
17. Lavrenko, V., Croft, W.B.: Relevance-based language models. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 120–127 (2001)

Time-Sensitive Language Modelling for Online Term Recurrence Prediction

Dell Zhang[1], Jinsong Lu[1], Robert Mao[2], and Jian-Yun Nie[3]

[1] Birkbeck, University of London
London WC1E 7HX, UK
dell.z@ieee.org, jingsong.lu@gmail.com
[2] Microsoft Corp.
Dublin, Ireland
robmao@microsoft.com
[3] University of Montreal
Quebec, H3C 3J7 Canada
nie@IRO.UMontreal.CA

Abstract. We address the problem of online term recurrence prediction: for a stream of terms, at each time point predict what term is going to recur next in the stream given the term occurrence history so far. It has many applications, for example, in Web search and social tagging. In this paper, we propose a time-sensitive language modelling approach to this problem that effectively combines term frequency and term recency information, and describe how this approach can be implemented efficiently by an online learning algorithm. Our experiments on a real-world Web query log dataset show significant improvements over standard language modelling.

1 Introduction

Consider a stream of terms[1] w_i with time stamp t_i: $(w_1, t_1), (w_2, t_2), \ldots$, where t_i monotonically increases. At each time point, we would like to predict what term is going to recur next in the stream, given the term occurrence history so far. This problem of *online term recurrence prediction* has many applications. For example, predicting the next query to be reused in a stream of queries is key to *query auto-completion* [1], *query suggestion* [2], *information re-finding* [3], and *result caching/prefetching* [4,5,6,7] in Web search engines, while predicting the next tag to be reused in a stream of tags is key to *tag auto-completion* [1] and *tag suggestion/recommendation* [8,9,10,11] in social tagging services.

The standard language modelling [12,13] approach to this problem relies solely on term frequency information to find the most probable next term to recur. However, our experience tells us that the terms occurred recently should have a higher probability of recurring than those occurred a long time ago, due to the

[1] Here terms are just language units which may contain one or more words, e.g., Web queries and social tags.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 128–138, 2009.

common phenomena of burst and drift in user interests [14]. For example, if a user searched ‘baseball’ yesterday and ‘basketball’ one month ago on the Web, she is more likely to search ‘baseball’ rather than ‘basketball’ again today. Such valuable information of term recency has been overlooked in standard language modelling.

In this paper, we propose a *time-sensitive language modelling* approach to this problem. It can effectively combine term frequency and term recency information. Furthermore, it can be implemented efficiently by an *online learning* [15] algorithm. Our experiments on a real-world Web query log dataset show that it brings significant improvements over standard language modelling.

The rest of this paper is organised as follows. In Section 2, we present our time-sensitive language modelling approach to the problem of online term recurrence prediction, and explain how it combines term frequency and term recency information effectively. In Section 3, we describe an online learning algorithm to implement the above approach efficiently. In Section 4, we empirically evaluate our technique on a real-world Web query log dataset. In Section 5, we review related work. In Section 6, we discuss future work. In Section 7, we make conclusions.

2 Approach

The basic idea of our time-sensitive language modelling technique is that each occurrence of a term will contribute to its probability of recurring in the future, but the amount or weight of contribution decays over time according to a *kernel function* $k(t, t_o)$. Here we use the *exponential decay* function as the kernel.

$$k(t, t_o) = \begin{cases} \exp(-\lambda(t - t_o)) & \text{if } t \geq t_o \\ 0 & \text{if } t < t_o \end{cases} ,$$

where $\lambda \geq 0$ is called the *decay constant*. The physical interpretation of this model is that one occurrence of term w at time t_o has an initial weight $W(t_o) = 1$ and it ‘evaporates’ at a rate proportional to its weight at that time $W(t)$: $dW/dt = -\lambda W$. The *mean lifetime*, i.e, the time needed for the initial weight to be reduced by a factor of e, is given by $\tau = 1/\lambda$.

So at time t, if the history so far is $H = \{(w_1, t_1), \ldots, (w_n, t_n)\}$, then the accumulated weight of a specific term w’s contribution can be calculated as

$$C(w, t) = \sum_{i=1}^{n} \delta(w, w_i) k(t, t_i) \ ,$$

where $\delta(w, w_i)$ is the Kronecker’s delta function: $\delta(w, w') = 1$ if $w = w'$ and 0 otherwise. The function $C(w, t)$ summarises the contribution of term w’s history to its occurring probability at time t, so we call it *contribution function.*

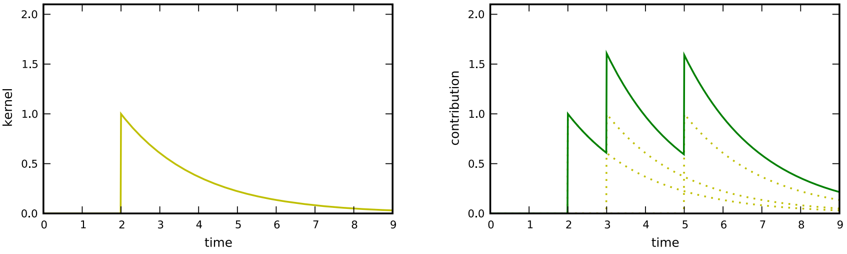
Figure 1 shows an example of time-sensitive language modelling based on exponential decay with $\lambda = 0.5$. In this example, the contribution of term w at time $t = 8$ will be

$$\begin{aligned} C(w,8) &= k(8,5) + k(8,3) + k(8,2) \\ &= \exp(-3\lambda) + \exp(-5\lambda) + \exp(-6\lambda) \\ &= \exp(-1.5) + \exp(-2.5) + \exp(-3) \\ &\approx 0.3550 \end{aligned}$$

Hence the probability of term w occurring at time t, given the history H, will be determined by its accumulated contribution at that time:

$$\widehat{P}(w_{n+1} = w | H, t_{n+1} = t) = \widehat{P}(w | C(w,t)) = \frac{C(w,t) + \mu}{\sum_{w'} [C(w',t) + \mu]} \ .$$

where $0 \le \mu \le 1$ is a parameter for Lidstone (additive) smoothing [16]. If $\mu = 0$, the above formula gives the Maximum Likelihood Estimation (MLE) of w's occurring probability that combines both its frequency and its recency information. However, for a term never occurred before in the history (or the training corpus), the MLE of its occurring probability will be 0, which is in general undesirable. This problem can be remedied by using a positive smoothing parameter μ, which can be regarded as a non-decaying constant weight of contribution assigned to every term (whether occurred before or not) by default.



(a) The kernel function $k(t,2)$.

(b) The contribution function $C(w,t)$ for a term that occurred at times 2, 3 and 5.

Fig. 1. An example of time-sensitive language modelling based on exponential decay with $\lambda = 0.5$

For the problem of online term recurrence prediction, what we need is to compute

$$w^* = \arg\max_w \widehat{P}(w_{n+1} = w | H, t_{n+1} = t) = \arg\max_w C(w,t) \ .$$

Now let's study the behaviour of time-sensitive language models in the context of online term recurrence prediction. Without loss of generality, we assume the system time is discrete, i.e., represented by an integer.

Proposition 1. *In a time-sensitive language model with decay constant $\lambda = 0$, the most probable next term to recur is the most frequently used (MFU) term.*

Proof. If $\lambda = 0$, then for each time point $t_i < t$ the kernel function $k(t, t_i) = exp(0) = 1$. So the contribution of term w at current time t is

$$C(w,t) = \sum_{i=1}^{n} \delta(w, w_i) k(t, t_i) = \sum_{i=1}^{n} \delta(w, w_i) \ ,$$

which is exactly the number of w's occurrences in the history. Therefore, the most probable next term $w^* = \arg\max_w C(w, t)$ is the term of highest frequency, i.e., the most frequently used (MFU) term. This also implies that when $\lambda = 0$, time-sensitive language modelling backs off to standard language modelling. □

Proposition 2. *In a time-sensitive language model with decay constant* $\lambda \geq \ln(2)$*, the most probable next term to recur is the most recently used (MRU) term.*

Proof. Let $r = \exp(-\lambda)$. Thus for each time point $t_i < t$ the kernel function can be written as $k(t, t_i) = \exp(-\lambda)^{t-t_i} = r^{t-t_i}$. Since $\lambda \geq \ln(2)$, we have $r \leq \exp(-\ln(2)) = \frac{1}{2}$. Consider two terms w_a and w_b that have occurred at times $T_a = \{t_{a_1}, \ldots, t_{a_n}\}$ and $T_b = \{t_{b_1}, \ldots, t_{b_n}\}$ respectively so far. Assume $t_{a_n} = u$ and $t_{b_n} = v$, and without loss of generality assume that $u < v$ or equivalently $u + 1 \leq v$ (as discrete time is used here).

$$\begin{aligned}
C(w_a, t) &= \sum_{i=1}^{n} \delta(w_a, w_i) k(t, t_i) = \sum_{t_i \in T_a} k(t, t_i) \\
&\leq \sum_{j=1}^{u} k(t, j) = \sum_{j=1}^{u} r^{t-j} = \frac{r^{t-(u+1)} - r^{t-1}}{1/r - 1} \\
&\leq r^{t-(u+1)} - r^{t-1} \\
&< r^{t-(u+1)} = k(t, u+1) \\
&\leq k(t, v) \ . \\
C(w_b, t) &= \sum_{i=1}^{n} \delta(w_b, w_i) k(t, t_i) = \sum_{t_i \in T_b} k(t, t_i) \\
&\geq k(t, v) \ .
\end{aligned}$$

To sum up, $C(w_a, t) < C(w_b, t)$ as long as $t_{a_n} < t_{b_n}$, i.e., the contribution of a term is dominated by its last occurrence time. Therefore, the most probable next term $w^* = \arg\max_w C(w, t)$ is the term of highest recency, i.e., the most recently used (MRU) term. □

The above two propositions show that at one extreme ($\lambda = 0$) the time-sensitive language modelling approach to online term recurrence prediction subsumes the approach of selecting the MFU terms (as in standard language modelling), while at the other extreme ($\lambda = \ln(2)$) it subsumes the approach of selecting the MRU terms. With a non-trivial decay constant $0 < \lambda < \ln(2)$, the time-sensitive language modelling approach differs from the MFU approach in that the contribution of each occurrence is not always the same but depends on its recency; it also differs from the MRU approach in that it considers not only the most recent occurrence, but also all the other past occurrences in the history. The decay constant λ controls the trade-off between term frequency and term recency in projecting a term's probability of being used in the future. What value of λ is optimal depends on the concrete problem and data.

3 Algorithm

A naive implementation of time-sensitive language modelling would be computationally expensive, because for each distinctive term, (1) we need to retain all its past occurrences for calculating its current contribution, and also (2) we need to constantly re-calculate its current contribution when time goes by. However, both issues turn out to be avoidable.

Theorem 1. *If there is no occurrence of term w in the period of time $[t_u, t_v]$, then its contribution satisfies*

$$C(w, t_v) = \exp(-\lambda(t_v - t_u))C(w, t_u) \ .$$

Proof. The contribution of term w at time t_u is

$$C(w, t_u) = \sum_{1 \le t_i \le t_u} \delta(w, w_i) k(t_u, t_i) \ ,$$

while the contribution of term w at time $t_v > t_u$ is

$$\begin{aligned} C(w, t_v) &= \sum_{1 \le t_i \le t_v} \delta(w, w_i) k(t_v, t_i) \\ &= \sum_{1 \le t_i \le t_u} \delta(w, w_i) k(t_v, t_i) + \sum_{t_u < t_i \le t_v} \delta(w, w_i) k(t_v, t_i) \ . \end{aligned}$$

As there is no occurrence of w between t_u and t_v, $\delta(w, w_i) = 0$ for all times t_i that $t_u < t_i \le t_v$. So we have

$$\begin{aligned} C(w, t_v) &= \sum_{1 \le t_i \le t_u} \delta(w, w_i) k(t_v, t_i) + \sum_{t_u < t_i \le t_v} 0 \cdot k(t_v, t_i) \\ &= \sum_{1 \le t_i \le t_u} \delta(w, w_i) k(t_v, t_i) \\ &= \sum_{1 \le t_i \le t_u} \delta(w, w_i) \exp(-\lambda(t_v - t_i)) \\ &= \sum_{1 \le t_i \le t_u} \delta(w, w_i) \exp(-\lambda((t_v - t_u) + (t_u - t_i)) \\ &= \exp(-\lambda(t_v - t_u)) \sum_{1 \le t_i \le t_u} \delta(w, w_i) \exp(\lambda(t_u - t_i)) \\ &= \exp(-\lambda(t_v - t_u)) C(w, t_u) \ . \end{aligned}$$

□

Corollary 1. *The contribution of term w at the current time t can be calculated using its last occurrence (i.e., most recent occurrence) time t_l and its contribution at t_l:*

$$C(w,t) = \exp(-\lambda(t - t_l))C(w,t_l) \ .$$

Proof. Obviously there should be no occurrence of term w between its last occurrence time t_l and the current time t. So using the above theorem, we get this corollary straightforwardly. □

According to this corollary, we only need to retain two values, t_l and $C(w,t_l)$, for each distinctive term w in the system.

Corollary 2. *Suppose two different queries w_a and w_b last occurred at times t_{a_n} and t_{b_n} respectively. Let $t_m = \max(t_{a_n}, t_{b_n})$. At any time $t > t_m$ we have*

$$cmp(C(w_a,t), C(w_b,t)) = cmp(C(w_a,t_m), C(w_b,t_m)) \ ,$$

where cmp is the comparison function

$$cmp(x,x') = \begin{cases} +1 \ \mathit{if} \ x > x' \\ \ \ 0 \ \mathit{if} \ x = x' \\ -1 \ \mathit{if} \ x < x' \end{cases} \ .$$

Proof. Obviously there should be no occurrence of term w between the time t_m and the current time t. So using the above theorem, we get

$$C(w_a,t) = \exp(-\lambda(t - t_m))C(w_a,t_m)$$
$$C(w_b,t) = \exp(-\lambda(t - t_m))C(w_b,t_m) \ .$$

Hence their ratio $\frac{C(w_a,t)}{C(w_b,t)} = \frac{C(w_a,t_m)}{C(w_b,t_m)}$. So if $\frac{C(w_a,t_m)}{C(w_b,t_m)} > 1$, i.e., $C(w_a,t_m) > C(w_b,t_m)$, then $\frac{C(w_a,t)}{C(w_b,t)} > 1$, i.e., $C(w_a,t) > C(w_b,t)$. Similar conclusions can be drawn for the cases $\frac{C(w_a,t_m)}{C(w_b,t_m)} = 1$ and $\frac{C(w_a,t_m)}{C(w_b,t_m)} < 1$ as well. Thereby completing the proof. □

According to this corollary, although the contribution of each term changes over time, the relative order of two terms' contributions or their probabilities to occur next does not change, until either of them recurs. Therefore the only chance for another term to replace the current most probable next term w^* is when it occurs.

The above two corollaries enable us to cut the computational overhead of time-sensitive language modelling drastically. Furthermore, they make it possible to continuously update and apply a time-sensitive language model through *online learning* [15] so that it can deal with *stream data* [14].

Figure 2 shows the online term recurrence prediction algorithm using a time-sensitive language model. The *input* of this algorithm is a series of time-stamped term occurrences $\{(w_1,t_1),\ldots,(w_n,t_n)\}$ as well as the decay constant λ, and the *output* is a series of predictions for the most probable next term w^*. Hash table,

```
create a hash table h
$w^* =$ '␣'; $\mathtt{h}[w^*] = \langle 0, 0 \rangle$
for $1 \leq i \leq n$:
    if $w_i$ not in h:
        $C(w_i, t_i) = 1$
    else:
        $\langle t_l, C(w_i, t_l) \rangle = \mathtt{h}[w_i]$
        $C(w_i, t_i) = \exp(-\lambda(t_i - t_l)) C(w_i, t_l) + 1$
    $\mathtt{h}[w_i] = \langle t_i, C(w_i, t_i) \rangle$
    $\langle t_l^*, C(w^*, t_l^*) \rangle = \mathtt{h}[w^*]$
    $C(w^*, t_i) = \exp(-\lambda(t_i - t_l^*)) C(w^*, t_l^*)$
    if $C(w_i, t_i) > C(w^*, t_i)$:
        $w^* = w_i$
    output $w^*$
```

Fig. 2. The online term recurrence prediction algorithm using a time-sensitive language model

a data structure that associates keys with values, has been used in the algorithm to store for each distinctive term w its last occurrence time t_l and contribution $C(w, t_l)$. A hash table supports lookup, insertion and deletion of elements in O(1) time (i.e., constant time) on average [17]. It is easy to see that overall this algorithm has time complexity O(n) for a stream of n terms.

4 Experiments

We apply the proposed technique of time-sensitive language modelling to *personalised query recurrence prediction* in Web search: considering each user's search log as a stream of queries, at each time point predict what query will be reused by her based on her entire query history so far. We assume that each user has her own time-clock and it is incremented by one upon each query issued.

We use the AOL query log dataset [18] (that is provided to the research community by AOL search engine[2]) for our experiments. In this paper, we focus on the queries within the first week of March 2006. The queries have already been normalised through punctuation-removal and case-folding etc. Finally the query log dataset used in this paper consists of $1,908,135$ queries from $309,078$ users.

One point worth mentioning is that in this dataset, if a user requested the next "page" of results for some query, this appears as a subsequent identical query with a later time stamp. Therefore it is not possible for us to determine whether the user reused the last query or she just requested for more results for the same query. To avoid this systematic bias towards the MRU approach

[2] http://search.aol.com/

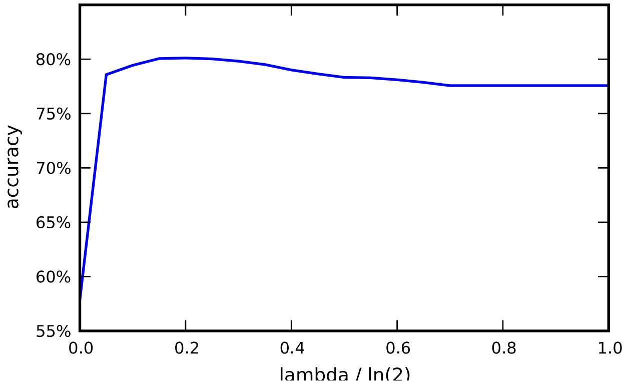


Fig. 3. The experimental results for personalised query recurrence prediction

in the experiments, we merge all identical successive queries from the same user into one. However, this implies that the immediate next query would always be different with the current one and the MRU approach would never succeed. To make a fair comparison between time-sensitive language modelling and standard language modelling, a small trick is applied here: we are actually making query recurrence predictions *two* steps forward, i.e., at time t_i we predict the query to be reused at time $t_i + 2$ by that user.

We evaluate the performance of our time-sensitive language modelling approach by comparing the top predicted queries with the later really occurred queries and computing the prediction accuracy. Here we report the accuracy of predicting recurred queries but not unseen queries, as the latter cannot be predicted based on the query history.

The experimental results for personalised query recurrence prediction using different decay constants are shown in Figure 3 and Table 1. The time-sensitive language modelling approach significantly outperforms the standard language modelling (i.e., MFU) approach. By using a non-zero decay constant to combine query frequency and query recency, we get an accuracy improvement of *more than a third*. The optimal decay constant $\lambda^* \approx 0.2 \times \ln 2 = 0.1386$. According to one-sided t-test [19,20], the effectiveness superiority of time-sensitive language modelling over standard language modelling is at the significance level 99.9% (P_value < 0.001).

To analyse the underlying reason for the success of time-sensitive language modelling, we rank all distinctive query recurrence incidents according to their frequency or recency values (in the corresponding user's individual query history) and calculate the proportion of query recurrence incidents for each frequency or recency rank. Figure 4 shows the *log-log* plots of query recurrence proportion over its frequency and recency rank respectively. We observe that in general, consistent with our intuition, (1) more *frequently* used queries are more likely to recur; and (2) more *recently* used queries are more likely to recur. We compute Kendall's rank-correlation coefficient τ [21] to quantitatively measure the utility

Table 1. The experimental results for personalised query recurrence prediction

language model	decay constant	prediction accuracy
standard	$\lambda = 0.0 \times \ln 2$ (MFU)	57.86%
time-sensitive	$\lambda = 0.1 \times \ln 2$	79.44%
	$\lambda = 0.2 \times \ln 2$	**80.11%**
	$\lambda = 0.3 \times \ln 2$	79.82%
	$\lambda = 0.4 \times \ln 2$	79.01%
	$\lambda = 0.5 \times \ln 2$	78.33%
	$\lambda = 0.6 \times \ln 2$	78.11%
	$\lambda = 0.7 \times \ln 2$	77.57%
	$\lambda = 0.8 \times \ln 2$	77.57%
	$\lambda = 0.9 \times \ln 2$	77.57%
	$\lambda = 1.0 \times \ln 2$ (MRU)	77.57%

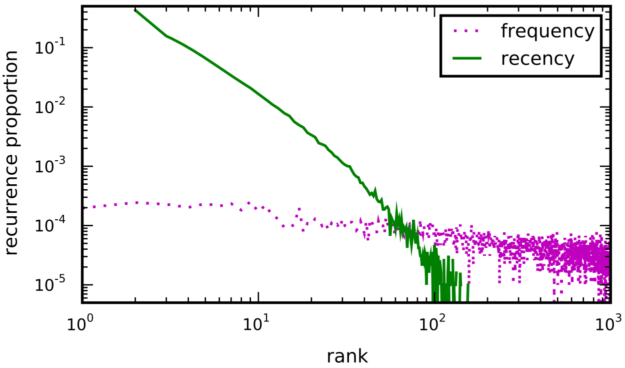


Fig. 4. The proportion of query recurrence over query frequency/recency

of query frequency and query recency in predicting query recurrence. The value of τ for the former is 0.4470 while that for the latter is 0.9367, which means that query recurrence is much more dependent on query recency than query frequency. Therefore the valuable information of query recency must not be discarded for personalised query recurrence prediction.

5 Related Work

The idea of making language models adaptive by introducing a decay function has appeared in various contexts such as speech recognition [22], news retrieval [23], email clustering [24], and collaborative filtering [25]. However, to the best of our knowledge, the effective behaviour and efficient implementation of time-sensitive language modelling for the problem of online term recurrence prediction have not been studied before.

6 Future Work

In this paper, we have focused on incorporating temporal decay into unigram language modelling, but it should be straightforward to extend our proposed technique to general n-gram language modelling [12,13] or even topic modelling [26]. It will also be interesting to investigate other temporal kernel functions in the proposed framework of time-sensitive language modelling, e.g., a periodic one to model the pattern of repetitive term occurrences.

7 Conclusions

The major contribution of this paper is the technique of time-sensitive language modelling that can address the problem of online term recurrence prediction effectively and efficiently.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments.

References

1. Garay-Vitoria, N., Abascal, J.: Text prediction systems: A survey. Universal Access in the Information Society 4(3), 188–203 (2006)
2. Mei, Q., Zhou, D., Church, K.W.: Query suggestion using hitting time. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM), Napa Valley, CA, USA, pp. 469–478 (2008)
3. Teevan, J., Adar, E., Jones, R., Potts, M.A.S.: Information re-retrieval: Repeat queries in yahoo's logs. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Amsterdam, The Netherlands, pp. 151–158 (2007)
4. Lempel, R., Moran, S.: Predictive caching and prefetching of query results in search engines. In: Proceedings of the 12th International World Wide Web Conference (WWW), Budapest, Hungary, pp. 19–28 (2003)
5. Baeza-Yates, R.A., Gionis, A., Junqueira, F., Murdock, V., Plachouras, V., Silvestri, F.: The impact of caching on search engines. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Amsterdam, The Netherlands, pp. 183–190 (2007)
6. Fagni, T., Perego, R., Silvestri, F., Orlando, S.: Boosting the performance of web search engines: Caching and prefetching query results by exploiting historical usage data. ACM Transactions on Information Systems (TOIS) 24(1), 51–78 (2006)
7. Gan, Q., Suel, T.: Improved techniques for result caching in web search engines. In: Proceedings of the 18th International Conference on World Wide Web (WWW), Madrid, Spain, pp. 431–440 (2009)
8. Heymann, P., Ramage, D., Garcia-Molina, H.: Social tag prediction. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Singapore, pp. 531–538 (2008)

9. Sigurbjornsson, B., van Zwol, R.: Flickr tag recommendation based on collective knowledge. In: Proceedings of the 17th International World Wide Web Conference (WWW), Beijing, China, pp. 327–336 (2008)
10. Song, Y., Zhuang, Z., Li, H., Zhao, Q., Li, J., Lee, W.C., Giles, C.L.: Real-time automatic tag recommendation. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Singapore, pp. 515–522 (2008)
11. Song, Y., 0007, L.Z., Giles, C.L.: A sparse gaussian processes classification framework for fast tag suggestions. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM), Napa Valley, CA, USA, pp. 93–102 (2008)
12. Manning, C., Schutze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge (1999)
13. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
14. Gaber, M.M., Zaslavsky, A.B., Krishnaswamy, S.: Mining data streams: A review. SIGMOD Record 34(2), 18–26 (2005)
15. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
16. Chen, S.F., Goodman, J.: An empirical study of smoothing techniques for language modeling. In: Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL), Morristown, NJ, USA, pp. 310–318. Association for Computational Linguistics (1996)
17. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. MIT Press/ McGraw-Hill (2001)
18. Pass, G., Chowdhury, A., Torgeson, C.: A picture of search. In: Proceedings of the 1st International Conference on Scalable Information Systems (Infoscale), Hong Kong, vol. 1 (2006)
19. Mitchell, T.: Machine Learning, international edn. McGraw Hill, New York (1997)
20. Yang, Y., Liu, X.: A re-examination of text categorization methods. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Berkeley, CA, pp. 42–49 (1999)
21. Kendall, M., Gibbons, J.D.: Rank Correlation Methods, 5th edn. A Charles Griffin Book (1990)
22. Clarkson, P.R., Robinson, A.J.: Language model adaptation using mixtures and an exponentially decaying cache. In: Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. 2, pp. 799–802 (1997)
23. Li, X., Croft, W.B.: Time-based language models. In: Proceedings of the 12th ACM Conference on Information and Knowledge Management (CIKM), New Orleans, LA, USA, pp. 469–475 (2003)
24. Zhu, X., Ghahramani, Z., Lafferty, J.: Time-sensitive dirichlet process mixture models. Technical Report CMU-CALD-05-104, Carnegie Mellon University (2005)
25. Ding, Y., Li, X.: Time weight collaborative filtering. In: CIKM, Bremen, Germany, pp. 485–492 (2005)
26. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research (JMLR) 3, 993–1022 (2003)

Score Distributions in Information Retrieval

Avi Arampatzis[1], Stephen Robertson[2], and Jaap Kamps[1]

[1] University of Amsterdam, the Netherlands
[2] Microsoft Research, Cambridge UK

Abstract. We review the history of modeling score distributions, focusing on the mixture of normal-exponential by investigating the theoretical as well as the empirical evidence supporting its use. We discuss previously suggested conditions which valid binary mixture models should satisfy, such as the Recall-Fallout Convexity Hypothesis, and formulate two new hypotheses considering the component distributions under some limiting conditions of parameter values. From all the mixtures suggested in the past, the current theoretical argument points to the two gamma as the most-likely universal model, with the normal-exponential being a usable approximation. Beyond the theoretical contribution, we provide new experimental evidence showing vector space or geometric models, and BM25, as being "friendly" to the normal-exponential, and that the non-convexity problem that the mixture possesses is practically not severe.

1 Introduction

Current best-match retrieval models calculate some kind of score per collection item which serves as a measure of the degree of relevance to an input request. Scores are used in ranking retrieved items. Their range and distribution varies wildly across different models making them incomparable across different engines [1], even across different requests on the same engine if they are influenced by the length of requests. Even most probabilistic models do not calculate the probability of relevance of items directly, but some order-preserving (monotone or isotone) function of it [2].

For single-collection ad-hoc retrieval, the variety of score types is not an issue; scores do not have to be comparable across models and requests, since they are only used to rank items per request per system. However, in advanced applications, such as distributed retrieval, fusion, or applications requiring thresholding such as filtering or recall-oriented search, some form of score normalization is imperative. In the first two applications, several rankings (with non-overlapping and overlapping sets of items respectively) have to be merged or fused to a single ranking. Here, score normalization is an important step [3]. In practice, while many users never use meta-search engines directly, most conventional search engines have the problem of combining results from many discrete sub-engines. For example, blending images, text, inline answers, stock quotes, and so on, has become common.

In filtering, bare scores give no indication on whether to retrieve an incoming document or not. Usually a user model is captured into some evaluation measure. Some of these measures can be optimized by thresholding the probability of relevance at some specific level [4], thus a method of normalizing scores into probabilities is needed.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 139–151, 2009.

Moreover, thresholding has turned out to be important in recall-oriented retrieval setups, such as legal or patent search, where ranked retrieval has a particular disadvantage in comparison with traditional Boolean retrieval: there is no clear cut-off point where to stop consulting results [5]. Again, normalizing scores to expected values of a given effectiveness measure allows for optimal rank thresholding. In any case, the optimal threshold depends on the effectiveness measure being used—there is no single threshold suitable for all purposes.

Simple approaches, e.g. range normalization based on minimum and maximum scores, are rather naive, considering the wild variety of score outputs across search engines, because they do not take into account the *shape* of score distributions (SDs). Although these approaches have worked reasonably well for merging or fusing results [6], advanced approaches have been seen which try to improve normalization by investigating SDs. Such methods have been found to work at least as well (or in some cases better than) the simple ones in the context of fusion [7,8]. They have also been found effective for thresholding in filtering [9,10,11] or thresholding ranked lists [12]. We are not aware of any empirical evidence in the context of distributed retrieval.

We review the history of modeling SDs in Information Retrieval, focusing on the currently most popular model, namely, the mixture of normal-exponential, by investigating the theoretical as well as the empirical evidence supporting its use. We discuss conditions which any valid—from an IR perspective—binary mixture model should satisfy, such as the Recall-Fallout Convexity Hypothesis, and formulate new hypotheses considering the component distributions individually as well as in pairs. Although our contribution is primarily theoretical, we provide new experimental evidence concerning the range of retrieval models that the normal-exponential gives a good fit, and try to quantify the impact of non-convexity that the mixture possesses. We formulate yet unanswered questions which should serve as directions for further research.

2 Modeling Score Distributions

Under the assumption of a binary relevance, classic attempts model SDs, on a per-request basis, as a mixture of two distributions: one for relevant and the other for non-relevant documents [13,14,15,16,17,7]. Given the two component distributions and their mix weight, the probability of relevance of a document given its score can be calculated straightforwardly [17,7], essentially allowing the normalization of scores into probabilities of relevance. Furthermore, the expected numbers of relevant and non-relevant documents above and below any rank or score can be estimated, allowing the calculation of precision, recall, or any other traditional measure at any given threshold enabling its optimization [12]. Assuming the right component choices, such methods are theoretically "clean" and non-parametric.

A more recent attempt models aggregate SDs of many requests, on per-engine basis, with single distributions [18,8]; this enables normalization of scores to probabilities—albeit not of relevance—comparable across different engines. The approach was found to perform better than the simple methods in the context of fusion [8]. Nevertheless, it is not clear—if it is even possible—how using a single distribution can be applied to thresholding, where for optimizing most common measures a reference to relevance is needed. For this reason, we will next concentrate on binary mixture models; moreover, we are not aware of any approach using SDs in beyond binary relevance setups.

Various combinations of distributions have been proposed since the early years of IR—two normal of equal variance [13], two normal of unequal variance or two exponential [14], two Poisson [15], two gamma [16]—with currently the most popular model being that of using a normal for relevant and an exponential for non-relevant, introduced in [9] and followed up by [17,7,10,11] and others. For a recent extended review and theoretical analysis of the above choices, we refer the reader to [1]. The latest improvements of the normal-exponential model use truncated versions of the component densities, trying to deal with some of its shortcomings [12]. Next we focus on the original normal-exponential model.

3 The Normal-Exponential Model

In this section, we review the normal-exponential model. We investigate the theoretical as well as the empirical evidence and whether these support its use.

3.1 Normal for Relevant

A theorem by Arampatzis and van Hameren [17] claims that the distribution of relevant document scores converges to a *Gaussian central limit* (GCL) quickly, with "corrections" diminishing as $O(1/k)$ where k is the query length. Roughly, three explicit assumptions were made:

1. Terms occur independently.
2. Scores are calculated via some linear combination of document term weights.
3. Relevant documents cluster around some point in the document space, with some hyper-ellipsoidal density (e.g. a hyper-Gaussian) with tails falling fast enough.

Next, we re-examine the validity and applicability of these assumptions in order to determine the range of retrieval models for which the theorem applies.

Assumption 1 is generally untrue, but see the further discussion below. Assumption 2 may hold for many retrieval models; e.g. it holds for dot-products in vector space models, or sums of partially contributing log-probabilities (log-odds) in probabilistic models. Assumption 3 is rather geometric and better fit to vector space models; whether it holds or not, or it applies to other retrieval models, is difficult to say. Intuitively, it means that the indexing/weighting scheme does its job: it brings similar documents close together in the document space. This assumption is reasonable and similar to the Cluster Hypothesis of K. van Rijsbergen [19, Chapter 3]. Putting it all together, the proof is more likely to hold for setups combining the following three characteristics:

- Vector space model, or some other geometric representation.
- Scoring function in the form of linear combination of document term weights, such as the dot-product or cosine similarity of geometric models or the sum of partially contributing log-probabilities of probabilistic models.
- Long queries, due to the convergence to a GCL depending on query length.

This does not mean that there exists no other theoretical proof applicable to more retrieval setups, but we have not found any in the literature.

A Note on Term Independence. Term independence assumptions are common in the context of probabilistic models and elsewhere, but are clearly not generally valid. This has elicited much discussion. The following points have some bearing on the present argument:

- Ranking algorithms derived from independence models have proved remarkably robust, and unresponsive to attempts to improve them by including dependencies.
- Making the independence assumption conditional on relevance makes it a little more plausible than a blanket independence assumption for the whole collection.
- Cooper [20] has shown that for the simple probabilistic models, one can replace the independence assumptions with linked dependence (that is, linked between the relevant and non-relevant sets), and end up with the same ranking algorithms. This may be a partial explanation for the robustness of the independence models.
- This linked dependence unfortunately does not help us with the present problem.
- Cooper et al. [21] show that if we want to estimate an explicit, well-calibrated probability of relevance for each document (to show to the user), then corrections need to be made to allow for the inaccuracies of the (in)dependence assumptions.

What these points emphasise is the very strong distinction between on the one hand having a scoring system which ranks well and on the other hand placing any stronger interpretation on the scores themselves.

3.2 Exponential for Non-relevant

Under a similar set of assumptions and approximations, Arampatzis and van Hameren [17] investigate also the distribution of non-relevant document scores and conclude that a GCL is unlikely and if it appears it does only at a very slow rate with k (practically never seen even for massive query expansion). Although such a theorem does not help much in determining a usable distribution, under its assumptions it contradicts Swets' use of a normal distribution for non-relevant [13,14].

The distribution in question does not necessarily have to be a known one. [17] provides a model for calculating numerically the SD of any class of documents (thus also non-relevant) using Monte-Carlo simulation. In absence of a related theory or a simpler method, the use of the exponential distribution has been so far justified empirically: it generally fits well to the *high-end* of non-relevant item scores, but not to all.

3.3 Normal-Exponential in Practice

The normal-exponential mixture model presents some practical difficulties in its application. Although the GCL is approached theoretically quickly as query length increases, practically, queries of length above a dozen terms are only possible through relevance feedback and other learning methods. For short queries, the Gaussian may simply not be there to be estimated. Empirically, using a vector space model with scores which were unbounded above on TREC data, [17] found usable Gaussian shapes to form at around $k = 250$. k also seemed to depend on the quality of a query; the better the query, the fewer the terms necessary for a normal approximation of the observed distribution.

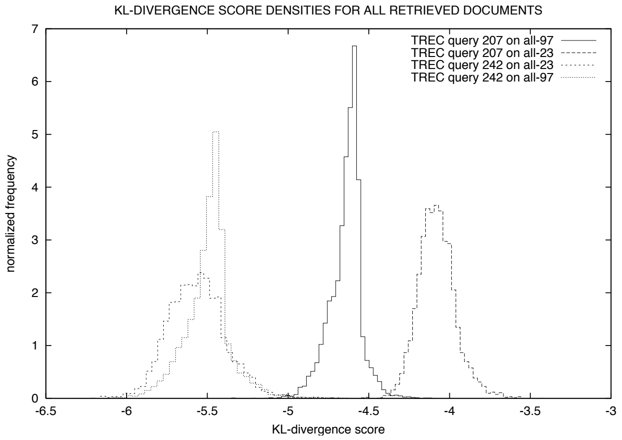


Fig. 1. KL-divergence score densities; two queries on two collections

Along similar lines, [7] noticed that better systems (in terms of average precision) produce better Gaussian shapes.

It was also shown in previous research that the right tail of the distribution of non-relevant document scores can be very well approximated with an exponential: [17,11] fit on the top 50–100, [7] fit on almost the top-1,000 (1,000 minus the number of relevant documents). [22] even fits on a non-uniform sample of the whole score range, but the approach seems system/task-specific. In general, it is difficult to fit an exponential on the whole score range. Figure 1 shows the total score densities produced by a combination of two queries and two sub-collections using KL-DIVERGENCE as a retrieval model. Obviously, none of these SDs can be fitted *in totality* with the mixture. Candidate ranges are, in general, $[s_{\text{peak}}, +\infty)$ where s_{peak} is set at the most frequent score or above.

Despite the above-mentioned practical problems, [7] used the model with success, with much shorter queries and even with a scoring system which produces scores between 0 and 1 without worrying about the implied truncation at both ends for the normal and at the right end for the exponential. In the context of thresholding for document filtering [11], with the generally unbounded scoring function BM25 and a maximum of 60 query terms per profile, the method performed well (2nd best, after Maximum Likelihood Estimation) on 3 out of 4 TREC data sets.

To further determine the retrieval models whose observed SDs can be captured well with a normal-exponential mixture, we investigated all 110 submissions to the TREC 2004 Robust track. This track used 250 topics combining the ad-hoc track topics in TRECs 6–8, with the robust track topics in TRECs 2003–2004. Table 1 shows the 20 submissions where the mixture obtained the best fit as measured by χ^2 goodness-of-fit test. The table shows the run names; the used topic fields; the median χ^2 upper probability indicating the goodness-of-fit; and the correlation between the optimal $F_1@K$ (with K a rank) based on the qrels and on the fitted distributions. The two remaining columns will be discussed in Section 4. Not surprisingly, over all runs, the 20 runs with the best fit also tend to have better predictions of $F_1@K$.

Table 1. Twenty submissions with the best normal-exponential goodness-of-fit

Run	Qry	χ^2	F_1 c.	NC	Inv.	Run	Qry	χ^2	F_1 c.	NC	Inv.
icl04pos2d	d	0.228	0.742	1.0	95.76	icl04pos2t	t	0.163	0.752	2.5	93.05
SABIR04FA	tdn	0.214	0.650	1.0	87.57	uogRobDWR10	d	0.158	0.642	1.0	89.35
icl04pos7f	tdn	0.197	0.663	2.0	93.64	wdo25qla1	tdn	0.157	0.579	4.0	83.12
icl04pos2f	tdn	0.190	0.629	1.0	93.66	icl04pos2td	td	0.154	0.718	1.0	95.87
SABIR04BA	tdn	0.185	0.658	1.0	90.25	uogRobLWR5	tdn	0.152	0.593	1.0	90.19
NLPR04OKapi	d	0.184	0.708	3.0	90.29	icl04pos7td	td	0.152	0.744	1.0	95.40
SABIR04FT	t	0.182	0.723	2.0	90.31	SABIR04BT	t	0.149	0.712	1.0	91.08
SABIR04FD	d	0.180	0.668	2.0	88.23	wdoqla1	tdn	0.149	0.637	2.0	85.66
SABIR04BD	d	0.174	0.647	2.0	88.05	uogRobDBase	d	0.148	0.646	1.0	88.31
icl04pos48f	tdn	0.166	0.694	1.0	95.78	fub04Dg	d	0.145	0.511	2.5	86.82

Looking at the retrieval models resulting in the best fits, we see seven runs of Peking University (icl) using a vector space model and the cosine measure. We also see 6 runs of Sabir Research, Inc. (SABIR) using the SMART vector space model. There are 3 runs of the University of Glasgow (uog) using various sums of document term weights in the DRF-framework. Two runs from Indiana University (wdo) using Okapi BM25. Finally, a single run from the Chinese Academy of Science (NLPR) using Okapi BM25, and one from Fundazione Ugo Bordoni (fub) also using sums of document term weights in the DRF-framework. Overall, we see support for vector space or geometrical models as being amenable to the normal-exponential mixture, as well as BM25.

Looking at query length, we see only 3 systems using the short title statement, and 8 systems using all topic fields. Many of the systems used query expansion, either using the TREC corpus or using the Web, leading to even longer queries. While longer queries tend to lead to smoother SDs and improved fits, the resulting $F_1@K$ prediction seems better for the short title queries with high quality keywords. The "pos2" runs of Peking University (icl) only index verbs and nouns, and considering only the most informative words seems to help distinguish the two components in the mixture.

4 The Recall-Fallout Convexity Hypothesis

From the point of view of how scores or rankings of IR systems should be, Robertson [1] formulates the Recall-Fallout Convexity Hypothesis:

> *For all good systems, the recall-fallout curve (as seen from the ideal point of recall=1, fallout=0) is convex.*

Similar hypotheses can be formulated as conditions on other measures, e.g., the probability of relevance should be monotonically increasing with the score; the same should hold for *smoothed* precision. Although, in reality, these conditions may not always be satisfied, they are expected to hold for good systems, i.e. those producing rankings satisfying the *probability ranking principle* (PRP), because their failure implies that systems can be easily improved.

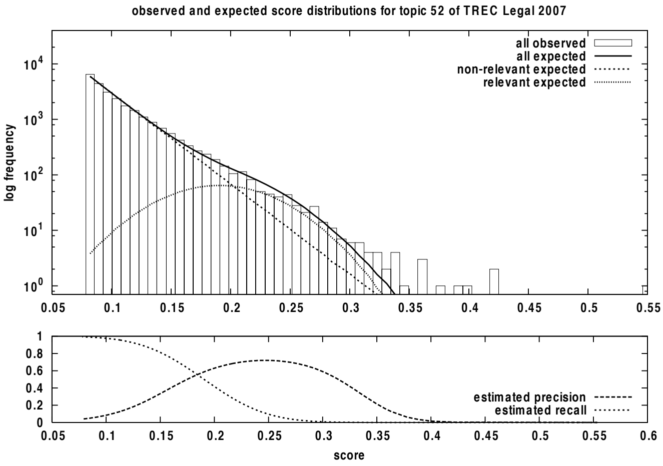


Fig. 2. Non-convexity inside the observed score range of a normal-exponential fit

As an example, let us consider smoothed precision. If it declines as score increases for a part of the score range, that part of the ranking can be improved by a simple random re-ordering [23]. This is equivalent of "forcing" the two underlying distributions to be uniform in that score range. This will replace the offending part of the precision curve with a flat one—the least that can be done—improving the overall effectiveness of the system. In fact, rankings can be further improved by reversing the offending sub-rankings; this will force the precision to increase with an increasing score, leading to better effectiveness than randomly re-ordering the sub-ranking.

Such hypotheses put restrictions on the relative forms of the two underlying distributions. Robertson [1] investigated whether the following mixtures satisfy the convexity hypothesis: two normals, two exponentials, two Poisson, two gamma, and normal-exponential. From this list, the following satisfy the hypothesis: two normal (only when their variances are equal), two exponential, two Poisson, and two gamma (for a quite wide range of parameters but not all).

Let us consider the normal-exponential mixture which violates such conditions only (and always) at both ends of the score range. Although the low-end scores are of insignificant importance, the top of the ranking is very significant. The problem is a manifestation of the fact that a normal falls more rapidly than an exponential and hence the two density functions intersect twice. Figure 2 depicts a normal-exponential fit on score data, together with the estimated precision and recall. The problem can be seen here as a declining precision above score 0.25.

In adaptive filtering, [9,22] deal with the problem by selecting as threshold the lower solution of the 2nd degree equation resulting from optimizing linear utility measures, while [10,11] do not seem to notice or deal with it. In meta-search, [7] noted the problem and *forced* the probability to be monotonic by drawing a straight line from the point where the probability is maximum to the point $[1, 1]$. Both procedures, although they may have been suitable for the above tasks, are theoretically unjustified. In [12], the two component distributions were set to uniform within the offending score range; as noted above, this is equivalent to randomization.

The problem does not seem severe for thresholding tasks. For example, [12] tried to optimize the F_1 measure and found that the impact of randomization on thresholding is that the SD method turns "blind" inside the offending range. As one goes down the corresponding ranks, estimated precision would be flat, recall naturally rising, so the optimal F_1 threshold can only be below the range. On average, the optimal rank threshold was expected to be deeper than the affected ranks, so the impact of non-convexity on thresholding deemed to be insignificant. Sometimes the problem may even appear above the maximum observed score. Furthermore, the truncated normal-exponential model used in [12] also helped to alleviate non-convexity by sometimes out-truncating it; a modest and conservative theoretical improvement over the original model which always violates the hypothesis.

To further determine the effect of the non-convexity of the normal-exponential, we again investigate the 110 submissions to the TREC 2004 Robust track. Table 1 also shows the median rank at which the estimated precision peaks (hence there is a non-convexity problem before this rank). We also show the effect of inverting the initial non-convex ranks, in percentage of overall MAP. That is, if precision increases up to rank 3 then it should make sense to invert the ranking of the first 2 documents. Two main observations are made. First, the median rank down to which the problem exists is very low, in the range of 1 (i.e. no practical problem) to 4, suggesting a limited impact on at least half the topics. Although there are outlier topics where the problem occurs far down the ranking, some of these may be due to problematic fits [12]. Second, "fixing" the problematic initial ranks by inverting the order leads to a loss of MAP throughout. This signals that the problem is not inherent in the underlying retrieval model violating the PRP. Rather, the problem is introduced by the fitted normal-exponential; both practical and fundamental problems can cause a misfit given the limited information available.

In the bottom line, the PRP dictates that any theoretically sound choice of component densities should satisfy the convexity condition; from all the mixtures suggested in the past, the normal-exponential as well as the normal-normal of unequal variances do not, for all parameter settings. In practice, the problem does not seem to be severe in the case of normal-exponential; the affected ranks are usually few. Given the theoretical and empirical evidence, we argue that the problem is introduced by the exponential, not by the normal. Moreover, many distributions—especially "peaky" ones—have a GCL. For example, assuming Poisson-distributed relevant document scores, for a system or query with a large mean score the Poisson would converge to a normal.

5 In-the-Limit Hypotheses

The Recall-Fallout Convexity Hypothesis considers the validity of *pairs* of distributions under the PRP. There are some reasons for considering distributions in pairs, as follows:

- The PRP is about the relative ranking of relevant and non-relevant documents under conditions of uncertainty about the classification; it makes no statements about either class in isolation.

- Consideration of the pair makes it possible for the hypothesis to ignore absolute scores, and therefore to be expressed in a form which is not affected by any monotonic transformation of the scores. Since ranking itself is not affected by such a transformation, this might be considered a desirable property.
- If we wish in the future to extend the analysis to multiple grades of relevance, a desirable general form would be a parametrised family of distributions, with different parameter values for each grade of relevance (including non-relevance), rather than a separately defined distribution for each grade.

However, the evidence of previous work suggests that the distributions of relevant and non-relevant look very different. This renders the third point above difficult to achieve, and further suggests that we might want to identify suitable hypotheses to apply to each distribution separately. Here we consider two hypotheses, the first of which achieves some degree of separation but may be difficult to support; the second is expressed in relative terms but may be more defensible.

Note that both hypothesis are "in the limit" conditions—they address what happens to the SDs under some limiting conditions of parameter values. They do not address the behaviour of distributions in other than these limiting conditions. Therefore they do not imply anything like the Recall-Fallout Convexity Hypothesis under actually observed parameter values.

5.1 The Strong Hypothesis

The ultimate goal of a retrieval system is not to produce some SD, but rather deliver the right items. In this light, the observed SD can be seen as an artifact of the inability of current systems to do a direct classification. Therefore, the ultimate SD all systems are trying to achieve is to the one with all relevant documents at the same high score $s_{\max}$, and all non-relevant documents at the same low score $s_{\min}$. The better the system, the better it should approximate the ultimate SD. This imposes restrictions on the two underlying components:

The Strong SD Hypothesis. *For good systems, the score densities of relevant and non-relevant documents should be capable of approaching Dirac's delta function, shifted to lie on the maximum score for the relevant and on the minimum score for the non-relevant, in some limiting condition.*

Let us now investigate which of the historically suggested distributions can approximate a delta and how.

The normal goes to delta via $\sigma \to 0$, and it can be positioned on demand via μ. The exponential approximates delta only via $\lambda \to +\infty$. The Poisson has one parameter λ, which incidentally equals both its mean and variance. For large λ, it approximates a normal with a mean and variance of λ. Consequently, as λ grows, the variance grows as well and it will never reach a delta. At the other side, for $\lambda = 0$ it becomes Kronecker's delta, i.e. the discrete analogue of Dirac's delta. The gamma has two parameters, $\Gamma(k, \theta)$. For large k it converges to a Gaussian with mean $k\theta$ and variance $k\theta^2$. The variance grows with k, but for $\theta \to 0$ it declines faster than the mean. So, the gamma can approximate a delta via an increasingly narrow Gaussian, and it can be positioned on demand via proper choices of k and θ.

Consequently, under the Strong SD Hypothesis, good candidates for relevant document scores are the normal or gamma, while for non-relevant are the normal, Poisson, exponential, or gamma. We only manage to reject the use of exponential and Poisson for relevant; although these could be simply shifted at $s_{\max}$ or vertically mirrored to end at $s_{\max}$, those setups would seem rather strange and unlikely.

Considering the historically suggested pairs of distributions, we can reject the mixture of two exponentials—at least as it was suggested in [14]: while the non-relevant exponential can approximate $\delta\left(s - s_{\min}\right)$ for $\lambda \to +\infty$, the relevant exponential cannot approximate $\delta\left(s - s_{\max}\right)$ for any λ. The two Poisson mixture of [15] is similarly rejected. The pairs remaining are the two normal, two gamma, or normal-exponential. Since a normal for non-relevant is unlikely according to [17] and Section 3.2, that leaves us with the two gamma or normal-exponential with only the former satisfying the convexity hypothesis for a range of parameter settings—not all. Note also that the two exponential or two Poisson constructions with the relevant component vertically mirrored would violate the Recall-Falout Convexity Hypothesis.

5.2 The Weak Hypothesis

The Strong SD Hypothesis would like to see all relevant documents at the same (high) score, and all non-relevant documents at the same (low) score. This requirement is not really compatible with any notion that there may actually be degrees of relevance (even if the user makes a binary decision), and is also not necessary for perfect ranking performance—either or both classes might cover a range of scores, provided only that they do not overlap. Thus we can formulate a weaker hypothesis:

The Weak SD Hypothesis. *For good systems, the score densities of relevant and non-relevant documents should be capable of approaching full separation in some limiting condition.*

Clearly, the Strong Hypothesis implies the Weak Hypothesis, because the Dirac delta function gives full separation.

The Weak Hypothesis, however, would not reject the mixture of two exponentials: as we push the mean of the non-relevant distribution down, non-relevant scores are increasingly concentrated around zero, while if we push the mean of the relevant distribution up, the relevant scores are more and more widely spread among high values. In the limit, perfect separation is achieved. The Weak Hypothesis also does not reject the Poisson mixture, if we achieve the limit by letting lambda go to zero for non-relevant and to infinity for relevant. This is similar to the mixture of two exponentials, except that the relevant scores are uniformly distributed over the positive integers only, instead of the positive real line.

The Weak Hypothesis is indeed weak, in that it does not reject any of the combinations previously discussed. However, it reveals significant differences in the notions of “perfect” retrieval effectiveness implicit in different combinations (and therefore what form improvements should take in SD terms). This “in the limit” behaviour is worth further exploration.

6 Conclusions and Directions for Future Research

The empirical evidence so far confirm that SD methods are effective for thresholding in filtering or ranked lists, as well as score normalization in meta-search. Specifically, the normal-exponential model seems to fit best vector space or geometric and BM25 retrieval models. Some mixtures have theoretical problems with an unclear practical impact. For example, using the normal-exponential model for thresholding the impact of non-convexity seems insignificant, however, elsewhere the effect may vary. Latest improvements of the model, namely, using truncated component densities alleviate the non-convexity problem—providing also better fits on data and better end-effectiveness in thresholding—without eliminating it [12].

The classic methods assume a binary relevance. A different approach would have to be taken, if degrees of relevance are assumed. For example, in TREC Legal 2008, there was a 3-way classification into non-relevant, relevant, and highly relevant. This complicates the analysis considerably, suggesting the need for three distributions. In this respect, it would fit more naturally with a model where both or all distributions came from the same family. It is difficult to see how one could adapt something like the normal-exponential combination to this situation. On the flip-side, approaches that analyze SDs without reference to relevance are just beginning to spring up [8]; nevertheless, these seem more suitable for score normalization for distributed IR or fusion rather than thresholding tasks.

An alternative approach would be to devise new scoring functions that have good distributional properties, or seek a calibration function by trying out different transformations on the scores of an existing system. Following the discussion on independence, we make a connection with the work of Cooper et al. [21], who argue that systems *should* give users explicit probability-of-relevance estimates, and use logistic regression techniques to achieve this. The idea of using logistic regression in this context dates back in [24], and re-iterated by others, e.g., [2]. The SD analysis indicates that in principle there should be such a calibration, which would take the form of a monotonic transformation of the score function, and therefore not affecting the ranking. Probability of relevance itself is sufficient for some of the thresholding tasks identified in the introduction but not for all—some require more complete distributional information. However, given probabilities of relevance we may find it easier to perform SD analysis and the chances of discovering a universal pair of distributions greater.

A universal pair should satisfy some conditions from an IR perspective. Although the two new hypotheses we introduced do not seem to align their demands with each other or with the older one, the pair that seems more “bullet-proof” is that of the two gamma suggested by [16]. The gamma can also become normal via a GCL or exponential via $k = 1$, thus allowing for the two exponential and normal-exponential combinations which are also likely depending on which conditions/hypotheses one considers. The increased degrees of freedom offered by the two gamma, however, is a two-edged sword: it may just allow too much. Parameter estimation methods introduce another layer of complexity, approximations, and new problems, as voiced by most previous experimental studies and more recently by [25]. At any rate, the distributions in question do not necessarily have to be known ones.

References

1. Robertson, S.: On score distributions and relevance. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 40–51. Springer, Heidelberg (2007)
2. Nottelmann, H., Fuhr, N.: From uncertain inference to probability of relevance for advanced IR applications. In: Sebastiani, F. (ed.) ECIR 2003. LNCS, vol. 2633, pp. 235–250. Springer, Heidelberg (2003)
3. Callan, J.: Distributed information retrieval. In: Advances Information Retrieval: Recent Research from the CIIR, pp. 127–150. Kluwer Academic Publishers, Dordrecht (2000)
4. Lewis, D.D.: Evaluating and optimizing autonomous text classification systems. In: Proceedings SIGIR 1995, pp. 246–254. ACM Press, New York (1995)
5. Oard, D.W., Hedin, B., Tomlinson, S., Baron, J.R.: Overview of the TREC 2008 legal track. In: Proceedings TREC 2008 (2009)
6. Lee, J.H.: Analyses of multiple evidence combination. In: Proceedings SIGIR 1997, pp. 267–276. ACM Press, New York (1997)
7. Manmatha, R., Rath, T.M., Feng, F.: Modeling score distributions for combining the outputs of search engines. In: Proceedings SIGIR 2001, pp. 267–275. ACM Press, New York (2001)
8. Fernández, M., Vallet, D., Castells, P.: Using historical data to enhance rank aggregation. In: Proceedings SIGIR 2006, pp. 643–644. ACM Press, New York (2006)
9. Arampatzis, A., Beney, J., Koster, C.H.A., van der Weide, T.P.: Incrementality, half-life, and threshold optimization for adaptive document filtering. In: Proceeding TREC 2000 (2000)
10. Zhang, Y., Callan, J.: Maximum likelihood estimation for filtering thresholds. In: Proceedings SIGIR 2001, pp. 294–302. ACM Press, New York (2001)
11. Collins-Thompson, K., Ogilvie, P., Zhang, Y., Callan, J.: Information filtering, novelty detection, and named-page finding. In: Proceedings TREC 2002 (2002)
12. Arampatzis, A., Robertson, S., Kamps, J.: Where to stop reading a ranked list? threshold optimization using truncated score distributions. In: Proceedings SIGIR 2009. ACM Press, New York (2009)
13. Swets, J.A.: Information retrieval systems. Science 141(3577), 245–250 (1963)
14. Swets, J.A.: Effectiveness of information retrieval methods. American Documentation 20, 72–89 (1969)
15. Bookstein, A.: When the most "pertinent" document should not be retrieved – an analysis of the Swets model. Information Processing and Management 13(6), 377–383 (1977)
16. Baumgarten, C.: A probabilitstic solution to the selection and fusion problem in distributed information retrieval. In: Proceedings SIGIR 1999, pp. 246–253. ACM Press, New York (1999)
17. Arampatzis, A., van Hameren, A.: The score-distributional threshold optimization for adaptive binary classification tasks. In: Proceedings SIGIR 2001, pp. 285–293. ACM Press, New York (2001)
18. Fernández, M., Vallet, D., Castells, P.: Probabilistic score normalization for rank aggregation. In: Lalmas, M., MacFarlane, A., Rüger, S.M., Tombros, A., Tsikrika, T., Yavlinsky, A. (eds.) ECIR 2006. LNCS, vol. 3936, pp. 553–556. Springer, Heidelberg (2006)
19. van Rijsbergen, C.J.: Information Retrieval, Butterworth (1979)
20. Cooper, W.S.: Some inconsistencies and misnomers in probabilistic information retrieval. In: Proceedings SIGIR 1991, pp. 57–61. ACM Press, New York (1991)
21. Cooper, W.S., Gey, F.C., Dabney, D.P.: Probabilistic retrieval based on staged logistic regression. In: Proceedings SIGIR 1992, pp. 198–210. ACM Press, New York (1992)

22. Arampatzis, A.: Unbiased s-d threshold optimization, initial query degradation, decay, and incrementality, for adaptive document filtering. In: Proceedings TREC 2001 (2002)
23. Robertson, S.E.: The parametric description of retrieval tests. part 1: The basic parameters. Journal of Documentation 25(1), 1–27 (1969)
24. Robertson, S.E., Bovey, J.D.: Statistical problems in the application of probabilistic models to information retrieval. Technical Report Report No. 5739, BLR&DD (1982)
25. Arampatzis, A., Kamps, J.: Where to stop reading a ranked list? In: Proceedings TREC 2008 (2008)

Modeling the Score Distributions of Relevant and Non-relevant Documents

Evangelos Kanoulas, Virgil Pavlu, Keshi Dai, and Javed A. Aslam*

College of Computer and Information Science
Northeastern University, Boston, USA
{ekanou,vip,daikeshi,jaa}@ccs.neu.edu

Abstract. Empirical modeling of the score distributions associated with retrieved documents is an essential task for many retrieval applications. In this work, we propose modeling the relevant documents' scores by a mixture of Gaussians and modeling the non-relevant scores by a Gamma distribution. Applying variational inference we automatically trade-off the goodness-of-fit with the complexity of the model. We test our model on traditional retrieval functions and actual search engines submitted to TREC. We demonstrate the utility of our model in inferring precision-recall curves. In all experiments our model outperforms the dominant exponential-Gaussian model.

1 Introduction

Information retrieval systems assign scores to documents according to their relevance to a user's request and return documents in a descending order of their scores. In reality, however, a ranked list of documents is a mixture of both relevant and non-relevant documents. For a wide range of retrieval applications (e.g. information filtering, topic detection, meta-search, distributed IR), *modeling* and *inferring* the distribution of relevant and non-relevant documents over scores in a reasonable way could be highly beneficial. For instance, in information filtering and topic detection modeling the score distributions of relevant and non-relevant documents can be utilized to find the appropriate threshold between relevant and non-relevant documents [16,17,2,19,7,15], in distributed IR it can be used for collection fusion [3], and in meta-search to combine the outputs of several search engines [10].

Inferring the score distribution for relevant and non-relevant documents in the absence of any relevance information is an extremely difficult task, if at all possible. *Modeling* score distributions in the right way is the basis of any possible inferences. Due to this, numerous combinations of statistical distributions have been proposed in the literature to model score distributions of relevant and non-relevant documents. In 60's and 70's Swets attempted to model the score distributions of non-relevant and relevant documents with two Gaussians

* We gratefully acknowledge the support provided by NSF grants IIS-0533625 and IIS-0534482.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 152–163, 2009.

of equal variance [16], two Gaussians of unequal variance and two exponentials [17]. Bookstein instead proposed a two Poisson model [6] and Baumgarten a two Gamma model [3]. A negative exponential and a Gamma distribution [10] has also been proposed in the literature. The dominant model, however, has been an exponential for the non-relevant documents and a Gaussian for the relevant ones [2,10,19].

As mentioned before the right choice of distributions (that is distributions that reflect the underline process that produces the scores of relevant and non-relevant documents) can enhance the ability to infer these distributions, while a bad choice may make this task practically impossible. Clearly a strong argument for choosing any particular combination of distributions is the goodness-of-fit to a set of empirical data. However, the complexity of the underline process that generates documents' scores makes the selection of the appropriate distributions a hard problem. Hence, even though the exponential - Gaussian model is the dominant one, there is no real consensus on the choice of the distributions. For instance, recently, Bennett [4], by utilizing the two Gaussians model for text classification and based on the observation that documents' scores outside the modes of the two Gaussians (corresponding to "extremely irrelevant" and "obviously relevant" documents) demonstrate different empirical behavior than the scores between the two modes (corresponding to "hard to discriminate" documents) introduced several asymmetric distributions to capture these differences.

Even though the goodness-of-fit can be a reasonable indicator of whether a choice of statistical distributions is the right one, from an IR perspective, these distributions should also possess a number of IR theoretical properties. Robertson considered various combinations of distributions and examined whether these combinations exhibit anomalous behavior with respect to theoretical properties of precision and recall [13].

In this work, we revisit the choice of distributions used to model documents' scores. Similarly to Bennett [4] we observed that the scores of relevant documents demonstrate different behavior in different score ranges. In order to study what is the appropriate choice of distributions for relevant and non-relevant documents we assume that the relevance information for all documents is available. We utilize a richer class of density functions for modeling the score distributions. In particular, we empirically fit a Gamma distribution in the scores of the non-relevant documents and a mixture of Gaussians in the scores of the relevant documents. Note that, the Gamma distribution represents the sum of M independent exponentially distributed random variables. In order to balance between the flexibility and the generalization power of the model we take a Bayesian treatment on the model that automatically trades-off the goodness-of-fit with the complexity of the model. We show that the data alone suggest that a mixture of two Gaussians for the relevant documents and a Gamma distribution with $M > 1$ is often times the right choice to model documents' scores. Further, we examine the IR utility of our model by testing how well one can infer precision-recall curves from the fit probability distributions. We show that our model outperforms the dominant exponential - Gaussian model.

2 Modeling Score Distributions

In this work, we empirically fit a Gamma distribution in the scores of the non-relevant documents and a mixture of Gaussians in the scores of the relevant documents (*GkG* model) and compare it to the dominant exponential-Gaussian model (*EF* model).

To avoid the effects of arbitrary query manipulations and score transformations that systems submitted to TREC (Text REtrieval Conference) often apply, in the sections that follow, we instead use scores produced by traditional IR models. Later, in Section 4, we validate our model on TREC systems.

The document collections used are the ones contained in TREC Disk 4 and 5, excluding the *Congressional Record* sub-collection, that is the exact same document collection used in TREC 8. The topics used are the TREC topics $401-450$ (the topics in TREC 8) [18]. Indexing and search was performed using the Terrier search engine [11]. Porter stemming and stop-wording was applied. The document scores obtained are the outputs of (a) Robertson's and Spärck Jones' TF-IDF [14], (b) BM25 [12], (c) Hiemstra's Language Model (LM) [9], and (d) PL2 divergence from randomness [1] (with Poisson estimation for randomness, Laplace succession for first normalization, and Normalization 2 for term frequency normalization). Further, three different topic formulations were used, (a) topic titles only, (b) topic titles and descriptions, and (c)topic titles, descriptions and narratives.

2.1 Methodology

The Gamma distribution was used to model the scores of the non-relevant documents. The Gamma density function with scale θ and shape M is given by,

$$P(x|M,\theta) = x^{M-1}\frac{\exp^{-M/\theta}}{\theta^M \Gamma(M)}$$

where, $\Gamma(M) = (M-1)!$. The mean of the distribution is $M\theta$, while the variance is $M\theta^2$. The maximum likelihood estimation (MLE) was used to estimate the Gamma parameters. When $M = 1$, the Gamma distribution degrades to an exponential distribution with rate parameter $1/\theta$.

The scores of relevant documents are modeled by a mixture of K Gaussians. Fitting the mixture of Gaussians into the scores could be easily done by employing the EM algorithm if the number of Gaussian components K was known. However, we considered as known only an upper bound on K. Given the fact that the larger the number of components is the better the fit will be and that EM finds the maximum likelihood mixture of Gaussians regardless of the model complexity, the EM algorithm is not appropriate for our problem. Instead, to avoid over-fitting, we employ a Bayesian treatment on the model by utilizing Variational Bayesian model selection for the mixture of Gaussians [5,8].

The mixture distribution of K Gaussian components is given by,

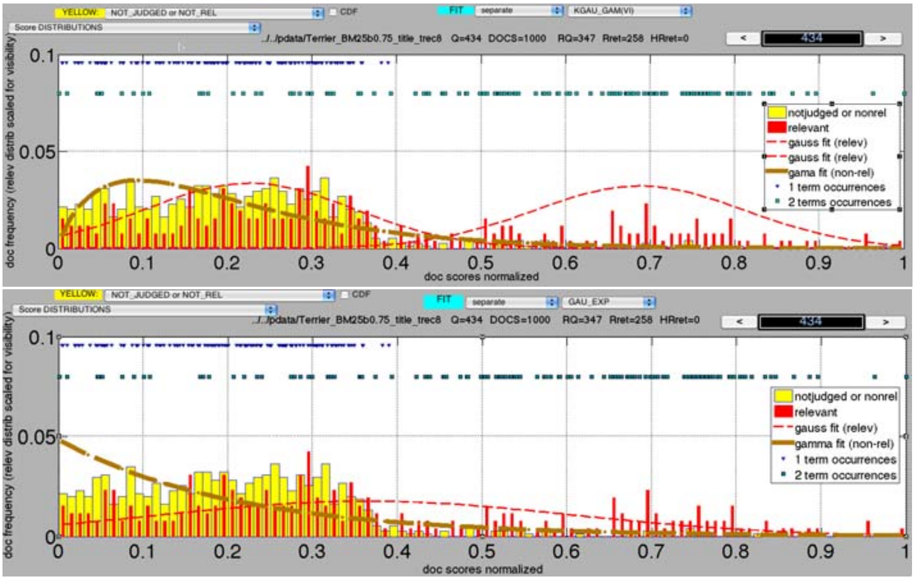
$$P(x|\pi,\mu,\Lambda) = \sum_{i=1}^{K} \pi_i \mathcal{N}(x|\mu_i,\Lambda_i^{-1})$$

where π_i are the mixing coefficients, and satisfy $0 \leq \pi_i \leq 1$ and $\sum_{i=1}^{K} \pi_i = 1$, μ_i and Λ_i the mean and the precision of the i^{th} Gaussian component.

The mixture coefficients π essentially give the contribution of each gaussian to the model. A fully Bayesian treatment of the mixture modeling problem involves the introduction of prior distributions over all the parameters, that is including π. Given a fixed number of potential components (an upper bound on K) the variational inference approach causes the mixing coefficients of unwanted components to go to zero and essentially leads to an automatic trade-off between the goodness-of-fit and the complexity of the model. The approach used in this paper to determine the number of components is to treat the mixing coefficients π as parameters and make point estimates of their value instead of maintaining a probability distribution over them [8].

2.2 Results and Analysis

We separately fit the Gamma distribution and the mixture of Gaussians into the scores of the non-relevant and relevant documents, respectively, per topic. There are 50 topics available and 3 query formulations (title, title and description and title, description and narrative), along with the relevance information for the top 1000 documents returned by 4 IR systems (TF-IDF, BM25, LM and PL2). Thus, there are in total 600 ranked lists of documents. The scores of the documents were first normalized into a 0 to 1 range.



YELLOW: NOT_JUDGED or NOT_REL □ CDF FIT separate KGAU_GAM(VI)
Score DISTRIBUTIONS
../../pdata/Terrier_BM25b0.75_title_trec8 Q=434 DOCS=1000 RQ=347 Rret=258 HRret=0

YELLOW: NOT_JUDGED or NOT_REL □ CDF FIT separate GAU_EXP
Score DISTRIBUTIONS
../../pdata/Terrier_BM25b0.75_title_trec8 Q=434 DOCS=1000 RQ=347 Rret=258 HRret=0

Fig. 1. The histogram over the scores of non-relevant and relevant documents and the Gamma and k Gaussians distribution (top) along with the negative exponential and single Gaussian distributions (bottom) fit into these scores separately

An example of fitting an exponential-Gaussian model and a Gamma and a mixture of two Gaussians into scores of non-relevant and relevant documents (separately) for query 434 ("Estonia economy") is shown in Figure 1. The wide yellow-bar and the thin red-bar histograms in both plots correspond to the non-relevant and relevant documents scores, respectively (scaled). Further, the top plot shows a negative exponential and a single Gaussian density functions fit into the scores, while the bottom plot shows a Gamma density function and a mixture of two Gaussians fit into the scores. As it can be observed, the Gamma and the mixture of two Gaussians can better fit the data than the choice of the exponential and the single Gaussian. To summarize our results we report the parameter M of the Gamma distribution, which as mentioned earlier corresponds to the number of independent exponential density functions averaged, and the number K of Gaussian components in the mixture, for all four systems, all 150 topics (50 topics and 3 query formulations). Figure 2 shows the histograms over M and K. As it can be observed, both M and K, in most of the cases, are different from 1, which shows that, taken into account the complexity of the model, the data suggest that a Gamma distribution and a mixture of Gaussians is a better fit than a negative exponential and a single Gaussian. In particular, the mean number of Gaussian components is 1.7, while the mean value of the parameter M is 1.3. In order to quantify and compare the goodness-of-fit for the different statistical distributions fit into the scores of relevant and non-relevant documents we employ hypothesis testing. The null hypothesis tested is that the scores of relevant (non-relevant) documents come from a certain distribution. The Kolmogorov-Smirnov test (using the maximum distance between the empirical and the theoretical cumulative distributions as a statistic) was utilized. The histogram of the p-values for all systems and all queries is shown in Figure 3. The top row corresponds to the p-values of testing the relevant documents scores against the single Gaussian distribution and mixture of K Gaussians, while the bottom row corresponds to the p-values of testing the non-relevant documents scores against the negative exponential and the Gamma distributions. As it can be observed, in the case of the relevant documents' scores distribution the single Gaussian distribution yields the worst results (as expected), with most of

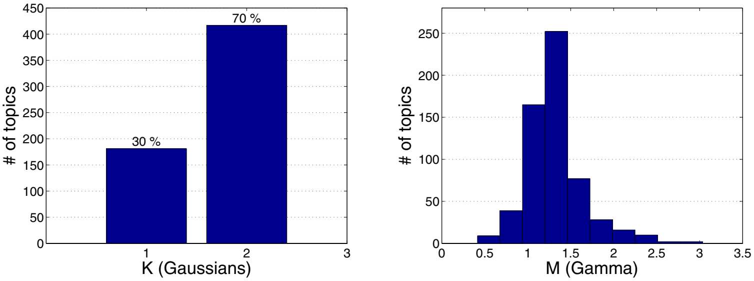


Fig. 2. The histograms over the number K of Gaussian components and the parameter M of the Gamma distribution, over all IR models, topics and topic formulations

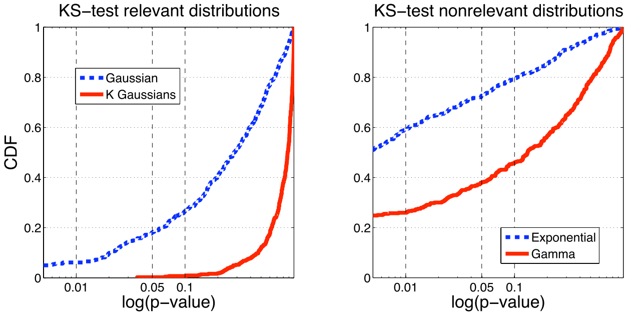


Fig. 3. The histogram of p-values of the Kolmogorov-Smirnov test on all systems, topics and topic formulations for relevant and non-relevant documents score distribution

the p-values being less than the significance level of 0.05 and thus rejecting the null hypothesis, while the mixture of two Gaussian distributions yields clearly much higher p-values. In particular, for 82% of the system-query pairs the null hypothesis that the score distribution is a single Gaussian distribution could not be rejected, while the corresponding percentage for the mixture of two Gaussians is **100%**. For the case of non-relevant documents the corresponding percentages for the exponential and Gamma distributions are 27% and **62%**, respectively.

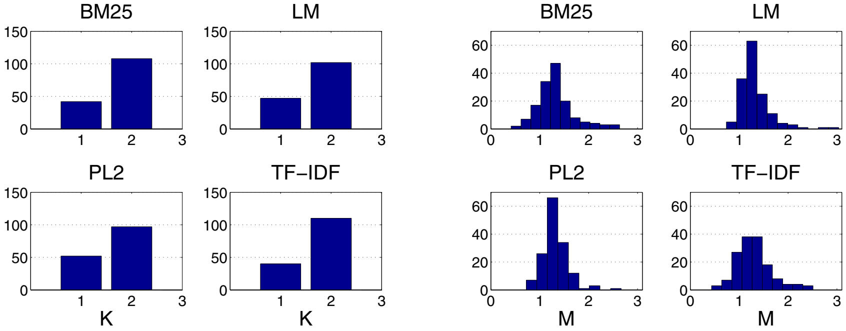


Fig. 4. The histogram over the number K of Gaussian components and the parameter M of Gamma distribution, over all topics and topic formulations for each IR model

Finally, we tested how the different IR systems and topic formulations affect the parameter M and the number K of Gaussian components. In Figures 4 and 5, we report the histograms over K for each system separately (50 topics with 3 topic formulations) and the histograms over K for each query formulation (all 50 topics and 4 IR systems). As it can be observed, the distribution of K appears to be independent both with respect to the IR model and with respect to query formulation. To validate our observations we run an n-way ANOVA testing whether the mean values of K per IR model - query formulation are equal and we could not reject the hypothesis.

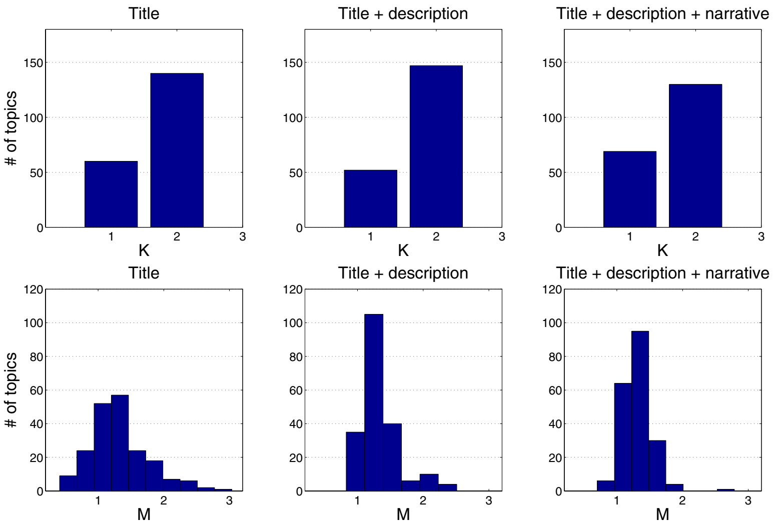


Fig. 5. The histogram over the number K of Gaussian components and the parameter M of Gamma distribution, over all topics and IR models for each topic formulation

2.3 On the Choice of Score Distributions

So far the optimal distributions to model the scores of relevant and non-relevant documents have been dictated by the data. In this section, we give an intuitive explanation of choice of a Gamma distribution to model non-relevant documents' scores and a mixture of Gaussians to model relevant documents' scores from an IR point of view.

An intuition behind the shape of the distribution that models the scores of relevant documents is given by Manmatha et al. [10]. Assuming that a query consists of a single term, Manmatha shows that the scores of relevant documents can be modeled as a Poisson distribution with a large λ parameter, which approaches a Gaussian distribution. Now, let's consider queries that consist of multiple terms and let's revisit the top plot in Figure 1. The query used in the example is: "Estonia economy". Each relevant document in the plot corresponds either to a triangular or to a rectangular marker at the top of the plot. The triangular markers denote the relevant documents for which only one out of the two query terms occur in the document, while the rectangular ones denote the relevant documents for which both terms occur in the document. By visual inspection, the relevant documents containing a single term clearly correspond to the low-scores' Gaussian, while the relevant documents containing both terms clearly correspond to the high-scores' Gaussian. Essentially, the former documents get a low score due to the fact that only one terms appear in them but they happen to be relevant to the query, while the latter correspond to documents that are obviously relevant. We observed the same phenomenon for many different queries independently of the IR model used for retrieval and independent of the query formulation. In the case of queries with multiple terms (e.g. queries that consists of both the title and the description), even though the possible number of query terms that may cooccur in a document is greater than 2 (e.g. for a query with 3 terms, all terms may occur in a document or only two of them or only a single one of them), we observed that there is a threshold on the number of terms occurring in the document; relevant documents containing a number of terms that is less than this threshold are clustered towards low scores (first Gaussian), while relevant documents containing a number of terms that is greater than the threshold are clustered towards high scores (second Gaussian).

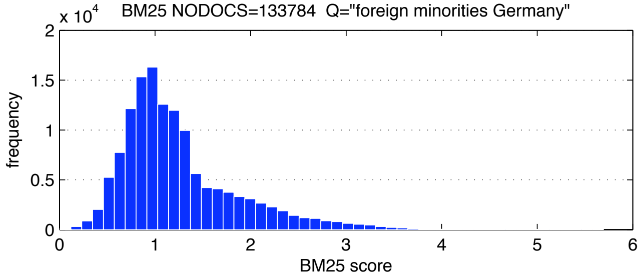


Fig. 6. The distribution of BM25 scores for all 133, 784 documents (containing at least one query term) on query "foreign minorities Germany". Note the different slopes at the left and at the right of the mean. Truncating the list at rank 1, 000 would cause the scores' distribution to look like an exponential one.

Regarding the non-relevant documents, given that the number of them is orders of magnitude larger than the number of the relevant ones, a modeling distribution over non-relevant documents' scores is essentially a modeling distribution over all scores. Previous work [10,13] argues that this distribution is a negative exponential but often times a more flexible distribution is necessary. The Gamma distribution, which can range (in skewness) from an exponential to a Gaussian distribution is flexible enough. In order to explain why a Gamma distribution is a better choice, several factors should be considered.

- Truncation cut-off: If a list is arbitrarily truncated very early (say at rank 1, 000) the distribution of the top scores may indeed look as an exponential. However looking deep down in the list (say up to rank 200, 000), the scores' distribution shape changes (Figure 6).
- Query complexity: Arguments for the scores' distribution for single term queries have been given in the literature [10]. For a query with two or more terms, most non-trivial documents (i.e. the ones that contain at least two query terms) will have the following property; the contribution of the two or more terms to the final score of a document would often times be very different for the two or more terms, with some terms having a low contribution while others having a higher contribution. Averaging such effects is likely to produce a "hill" of score frequencies, perhaps with different slopes

at the left and the right side of the mean; the Gamma distribution is known to be an average of exponential distributions.
- Retrieval function: We mostly look at scoring functions that are decomposable into a sum of scores per query terms, like TF-IDF or Language Models (after taking logs); such scores also induce averaging effects.

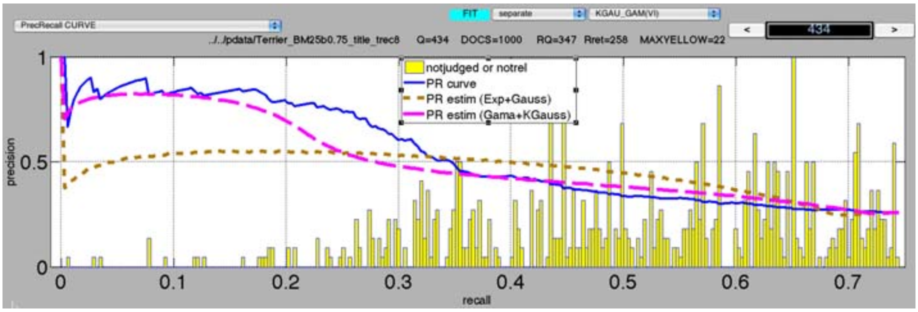


Fig. 7. Precision-Recall curve (blue) for query 434 and the BM25 retrieval function implemented by Terrier. It is easy to see that the PR curve estimated from the GkG model (magenta) is much better than the PR estimated from the EG model (brown). Yellow bars indicate the number of non-relevant documents in each recall interval.

3 Precision-Recall Curves

As a utility of our model for IR purposes, we estimate the precision-recall (PR) curve separately from both the *EG* and *GkG* model. Similarly to Robertson [13], let f_r and f_n denote the model densities of relevant and non-relevant scores, respectively; $F_r(x) = \int_x^1 f_r(x)dx$ and $F_n(x) = \int_x^1 f_n(x)dx$ are the cumulative density functions *from the right*. While the density models might have support outside the range [0,1], we use integrals up to 1 because our scores are normalized. For each recall level r we estimate the retrieval score at which r happens, from the relevant cumulative density: $score(r) = F_r^{-1}(r)$, which we compute numerically. Then we have $n(r) = F_n(score(r))$ as the percentage of non-relevant documents found up to recall r in the ranked list. Finally, the precision at recall r can be computed as in [13], $prec(r) = \frac{r}{r+n(r)*G}$, where G is the ratio of non-relevant to relevant documents in the collection searched. Computing precision at all recall levels from the score distribution models f_r and f_n gives an estimated PR curve. In the reminder of this section we show that estimating PR curves from the *GkG* model clearly outperforms PR curves estimated from the dominant *EG* model.

To measure the quality of the estimated PR curves we report the RMS error between the actual and the predicted precisions at all recall levels for both models. The results are summarized in Table 1, separately for each model. Language model LM and Divergence from randomness PL2 seem to produce slightly better PR estimates, independent of the query formulation. The over-all RMSE of *GkG* vs. *EG* is .094 vs .117, or about 20% improvement.

Table 1. RMS error between the actual and the inferred precision-recall curves

	title		title+desc		title+desc+narrative	
	EG	GkG	EG	GkG	EG	GkG
BM25	.135	.106	.122	.093	.117	.099
LM	.117	.098	.101	.085	.091	.076
PL2	.113	.092	.116	.094	.113	.092
TFIDF	.137	.106	.122	.095	.120	.100

Table 2. Mean Absolute Error between actual and inferred precision-recall curves

	title		title+desc		title+desc+narrative	
	EG	GkG	EG	GkG	EG	GkG
BM25	.091	.067	.076	.052	.071	.056
LM	.078	.063	.064	.052	.055	.043
PL2	.072	.056	.070	.052	.065	.049
TFIDF	.092	.067	.076	.053	.072	.055

Further, we report the mean absolute error between the actual and predicted precisions at all recall levels. This is the area difference between the estimated and the actual curve, which immediately gives a bound for the difference in Average Precision of the two curves (because the AP metric is approximated by the area under the PR curve). The results are reported in Table 2. Note that the best fit with respect to MAE are given for the full query formulation (title, description and narrative); the overall MAE for *GkG* is .055 vs *EG* with .074, or an improvement of about 25%.

4 TREC Search Engines

To avoid the effects of arbitrary query manipulations and score transformations that systems submitted to TREC (Text REtrieval Conference) often applied, we used in our experiments scores produced by traditional IR models. In this section we apply our methodology over the score distributions returned by search engines

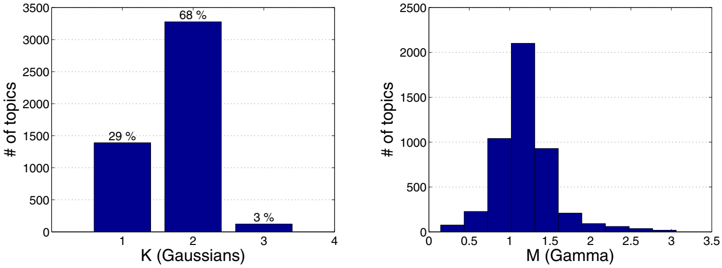


Fig. 8. The histograms over the number K of Gaussian components and the parameter M of the Gamma distribution, over all IR models, topics and topic formulations

submitted to TREC 8. Out of the 129 manual and automatic systems submitted to TREC 8 30 of them were excluded from our experiments since they transform document scores into ranks. No other quality control was performed. As earlier, we report the parameter M of the Gamma distribution, and the number K of Gaussian components in the mixture, for all systems and all queries as histograms in Figure 8. As it can be observed, similarly to the case of the traditional IR models, both M and K, in most cases, are different from 1, confirming that a Gamma distribution and a mixture of Gaussians is a better fit than a negative exponential and a single Gaussian.

5 Conclusions

In this work, we proposed modeling the relevant documents' scores by a mixture of Gaussians and modeling the non-relevant scores by a Gamma distribution. In all experiments conducted our model outperformed the dominant exponential-Gaussian model. Further, we demonstrated the utility of our model in inferring precision-recall curves. Some intuition about the choice of the particular model from an IR perspective was also given.

References

1. Amati, G., Van Rijsbergen, C.J.: Probabilistic models of information retrieval based on measuring divergence from randomness. ACM Transactions on Information Systems 20(4), 357–389 (2002)
2. Arampatzis, A., van Hameran, A.: The score-distributional threshold optimization for adaptive binary classification tasks. In: SIGIR 2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 285–293. ACM Press, New York (2001)
3. Baumgarten, C.: A probabilistic solution to the selection and fusion problem in distributed information retrieval. In: SIGIR 1999: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 246–253. ACM Press, New York (1999)
4. Bennett, P.N.: Using asymmetric distributions to improve text classifier probability estimates. In: SIGIR 2003: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, pp. 111–118. ACM Press, New York (2003)
5. Bishop, C.M.: Pattern Recognition and Machine Learning, Information Science and Statistics. Springer, Heidelberg (2006)
6. Bookstein, A.: When the most "pertinent" document should not be retrieved—an analysis of the swets model. Information Processing & Management 13(6), 377–383 (1977)
7. Collins-Thompson, K., Ogilvie, P., Zhang, Y., Callan, J.: Information filtering, novelty detection, and named-page finding. In: Proceedings of the 11th Text Retrieval Conference (2003)
8. Corduneanu, A., Bishop, C.M.: Variational bayesian model selection for mixture distributions. In: Proceedings Eighth International Conference on Artificial Intelligence and Statistics, pp. 27–34. Morgan Kaufmann, San Francisco (2001)

9. Hiemstra, D.: Using language models for information retrieval. PhD thesis, Centre for Telematics and Information Technology. University of Twente (2001)
10. Manmatha, R., Rath, T., Feng, F.: Modeling score distributions for combining the outputs of search engines. In: SIGIR 2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 267–275. ACM, New York (2001)
11. Ounis, I., Lioma, C., Macdonald, C., Plachouras, V.: Research directions in terrier. In: Baeza-Yates, R., et al. (eds.) Novatica/UPGRADE Special Issue on Next Generation Web Search, vol. 8(1), pp. 49–56 (2007) (invited Paper)
12. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: SIGIR 1994: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 232–241. Springer, New York (1994)
13. Robertson, S.: On score distributions and relevance. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 40–51. Springer, Heidelberg (2007)
14. Robertson, S.E., Jones, S.K.: Relevance weighting of search terms. Journal of the American Society for Information Science 27(3), 129–146 (1976)
15. Spitters, M., Kraaij, W.: A language modeling approach to tracking news events. In: Proceedings of TDT workshop 2000, pp. 101–106 (2000)
16. Swets, J.A.: Information retrieval systems. Science 141(3577), 245–250 (1963)
17. Swets, J.A.: Effectiveness of information retrieval methods. American Documentation 20, 72–89 (1969)
18. Voorhees, E.M., Harman, D.K.: TREC: Experiment and Evaluation in Information Retrieval. Digital Libraries and Electronic Publishing/ MIT Press (September 2005)
19. Zhang, Y., Callan, J.: Maximum likelihood estimation for filtering thresholds. In: SIGIR 2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 294–302. ACM, New York (2001)

Modeling Expected Utility of Multi-session Information Distillation

Yiming Yang and Abhimanyu Lad

Language Technologies Institute,
Carnegie Mellon University,
Pittsburgh, USA
{yiming,alad}@cs.cmu.edu

Abstract. An open challenge in information distillation is the evaluation and optimization of the utility of ranked lists with respect to flexible user interactions over multiple sessions. Utility depends on both the relevance and novelty of documents, and the novelty in turn depends on the user interaction history. However, user behavior is non-deterministic. We propose a new probabilistic framework for stochastic modeling of user behavior when browsing multi-session ranked lists, and a novel approximation method for efficient computation of the expected utility over numerous user-interaction patterns. Using this framework, we present the first utility-based evaluation over multi-session search scenarios defined on the TDT4 corpus of news stories, using a state-of-the-art information distillation system. We demonstrate that the distillation system obtains a 56.6% utility enhancement by combining multi-session adaptive filtering with novelty detection and utility-based optimization of system parameters for optimal ranked list lengths.

Keywords: Multi-session distillation, utility evaluation based both on novelty and relevance, stochastic modeling of user browsing behavior.

1 Introduction

Information distillation is an emerging area of research where the focus is to effectively combine ad-hoc retrieval (IR), novelty detection (ND) and adaptive filtering (AF) over temporally ordered documents for global utility optimization [12,2,11]. An information distillation system is typically designed for use over multiple sessions by a user or analyst. In each session, the system processes a new chunk of documents and presents a ranked list of passages[1] based on the utility of the passages to the user, where utility is measured in terms of *relevance* as well as *novelty*. The novelty of each passage in turn depends on the history of user interaction with the system, i.e., which passages were already seen by the user in the past. User behavior is typically non-deterministic, i.e., not every document in the system-produced ranked lists is necessarily read by the

[1] We use “passage” as a generic term for any retrieval unit, e.g., documents, paragraphs, sentences, etc.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 164–175, 2009.

user. They may skip passages, or abandon going further down a ranked list after reading the top few passages due to various reasons, e.g. satisfaction, frustration, and so on. The nondeterministic nature of user-browsing behavior has raised an important question – how should the expected utility of a distillation system be defined, estimated and maximized over all plausible patterns of user interactions in multi-session distillation? Current literature in IR, ND and AF has not offered a satisfactory solution for the whole problem, but only partial answers for subproblems.

Recently, there has been increasing interest in evaluation metrics that are based on a model of user behavior. For example, Moffat et al. proposed *Rank-Biased Precision* (RBP) [8], which corresponds to the expected rate of gain (in terms of graded relevance) obtained by a user who reads a ranked list *top down*, and whose stopping point in a ranked list is assumed to follow a geometric distribution. Similarly, Robertson et al. re-interpreted *Average Precision* as the expected precision observed by a user who stops with uniform probability at one of the relevant documents in the ranked list returned by the system [9]. To evaluate retrieval systems in multi-session search scenarios, Järvelin et al. proposed an extension to the Discounted Cumulated Gain (DCG) metric, known as session-based DCG (sDCG) [7] that discounts relevant results from later retrieval sessions[2], to favor early retrieval of relevant information in multi-session search scenarios, based on the assumption that examining retrieved results and reformulating the query involves an effort on the part of the user.

However, all these metrics are designed for measuring utility purely in terms of *relevance* – binary or graded. In many retrieval settings, especially scenarios involving multiple search sessions, *novelty* of information plays a crucial role in determining the overall utility of the system. Adding novelty to the definition of traditional IR metrics is not straight-forward, mainly due to its dynamic nature. Unlike *relevance*, which can be "pre-defined" for each document-query pair, novelty is an ever-changing function of which passages were read or skipped by the user in his or her interactions with the system up to the current point. Therefore, we cannot measure novelty without accounting for the dynamic and non-deterministic nature of user interaction.

Nevertheless, most novelty detection approaches and benchmark evaluations conducted in NIST and TREC have shared a convention of producing novelty judgments in an *offline* manner – all the passages which are relevant to a query are listed in a pre-specified order, and a binary judgment about the novelty of each passage is made, based on how its content differs from previous passages in the list [1]. Such novelty judgments would be correct from a user's perspective only if *all* these passages were presented by the system to the user, and in the exact same order as they were laid out during the ground truth assignment. These conditions may not hold in realistic use of a distillation system, which could show both relevant and non-relevant passages to the user, ranked according to its own notion of "good" passages.

[2] A note on terminology – In this paper, a *distillation task* consists of multiple *search sessions*, each comprising a single query. In [7], a *session* consists of multiple *queries.*

In other words, conventional evaluation schemes for novelty detection are insufficient or inappropriate for evaluating the true novelty – and hence – the true utility (relevance plus novelty) of passages in realistic settings. Non-deterministic user interactions over multi-session ranked lists make the problem even harder. The novelty of each passage would depend not only on the the user history in the current session, but also the user history in all previous sessions. Since there are many possible ways for the user to interact with multi-session ranked lists, we must evaluate the *expected* utility of the system over all interaction patterns instead of assuming a fixed pattern of user interaction, e.g., "all users read the top 10 passages in each ranked list." A principled solution would be to create a stochastic model of user behavior, and define a probability distribution over user interaction patterns with respect to multiple ranked lists, and accordingly calculate the expected utility of the system.

The above challenge has been partially addressed by the NDCU (Normalized Discounted Cumulated Utility) scheme proposed by Yang et al. [12] for distillation evaluation. NDCU uses nugget-level relevance judgments to enable automated determination of relevance and novelty of each passage in a system-produced ranked list. The evaluation algorithm scans the ranked list from top to bottom, keeping a count of all nuggets seen in each passage, thus dynamically updating the novelty of each nugget as the evaluation proceeds. Despite these desirable properties, a major limitation of NDCU is that it is only well-defined for a single ranked list. In case of a K-session distillation process, when estimating the novelty of passages in the k^{th} ranked list, how should "user history" at that point be modeled? Should we assume that all the ranked lists in the past $k-1$ sessions were completely read by the user? This assumption is obviously unrealistic. Alternatively, if we assume that the previous ranked lists were only partially browsed, then we need a principled way to model all plausible user-interaction patterns, and to estimate the *expected* utility of the system as a function of the joint probabilistic distribution of user interaction patterns over multiple sessions.

A recent approach by Clarke et al. [4] is similar to NDCU in terms of counting both relevance and novelty in utility-based evaluation and with respect to exploiting nugget-level relevance judgments. However, it also shares the same limitation with NDCU, namely not modeling stochastic user interaction patterns in multi-session distillation. sDCG [7] accommodates multiple search sessions, but lacking a probabilistic model of user behavior, it cannot account for the non-determinism associated with which passages were read by the user in each ranked list. Therefore, one is forced to make deterministic assumptions about user behavior, e.g., "users read a fixed number of documents in each ranked list" (the authors truncate each retrieved list at rank 10). Therefore, sDCG does not accurately reflect the true utility perceived by a user who can flexibly interact with multiple ranked lists presented by the system.

Our focus in this paper is to address the limitations of current methods for utility-based evaluation and optimization of distillation systems. Specifically, (i) We propose a new framework for probabilistic modeling of user browsing patterns over multi-session ranked lists. Each pattern corresponds to a possible way for a user to browse through the ranked lists. By summing over all such patterns, we calculate the Expected Global Utility of the system. (ii) This model flexibility comes at the cost of increased computational complexity, which we address using an efficient approximation technique. (iii) Using this framework, we present the first utility-based evaluation of a state-of-the-art distillation system, which produces ranked lists based on relevance as well as novelty of passages. By comparing different configurations of the system, we demonstrate that the proposed evaluation metric can effectively measure the utility of ranked lists in terms of relevance, novelty, as well as the reading cost of presented information.

We start by briefly describing the NDCU evaluation metric in the next section, followed by detailed explanation of the new framework.

2 Normalized Discounted Cumulated Utility

The Normalized Discounted Cumulated Utility (NDCU) scheme [12] is an extension of the popular NDCG metric [6] to model utility as the difference between the gain and cost incurred by a user in going through a ranked list presented by the system. Specifically, the utility of each passage is defined as:

$$U\left(p_i|l_q\right) = G\left(p_i|l_q\right) - aC(p_i) \tag{1}$$

where q is a query, l_q is the ranked list retrieved for the query, p_i is i^{th} passage in l_q, $G\left(p_i|l_q\right)$ is the *gain* (benefit) for reading the passage, $C(p_i)$ is the cost for reading the passage, and a is a pre-specified constant for balancing the gain and the cost of user interaction with the passage. The cost for reading the passage is defined as the passage length in terms of the number of words. The gain from reading the passage is defined in terms of its relevance and novelty, as follows:

$$G\left(p_i|l_q\right) = \sum_{\delta \in p_i} w(\delta, q)\gamma^{n(\delta, l_q, i-1)} \tag{2}$$

where δ is a nugget (a unit for relevance judgment), $w(\delta, q)$ is the graded relevance of δ with respect to the query q, $n(\delta, l_q, i-1)$ is the number of times δ appears in the ranked list l_q up to rank $i-1$. γ is a pre-specified *dampening factor*, reflecting the user's tolerance for redundancy. If $\gamma = 1$, the user is assumed to be fully tolerant to redundancy, and the evaluation reduces to be relevance-based only. At the other extreme of $\gamma = 0$, reading a nugget after the first time is assumed to be totally useless for the user, and hence incurs only cost. The use of nuggets as retrieval units allows flexible evaluation over arbitrary system output, as well as fine-grained determination of novelty.

The Discounted Cumulated Utility (DCU) of a list l_q is calculated as:

$$DCU(l_q) = \sum_{i=1}^{|l_q|} P(R_i = 1)\left(G\left(p_i|l_q\right) - aC(p_i)\right) \tag{3}$$

where $|l_q|$ is the number of passages in the ranked list, and $P(R_i = 1)$ is the probability that the passage with rank i in the list is read by the user. Since

$P(R_i = 1)$ is typically a decreasing function of the rank, it serves as a discounting factor, similar to the logarithmic discount used in DCG [6]. The DCU score of the system can be normalized by the DCU score of the ideal ranked list to obtain Normalized Discounted Cumulated Utility (NDCU).

Combining relevance and novelty into a utility-based evaluation metric, and utilizing nugget-level judgments to enable automated calculation of novelty for passages in any ranked list, were the main accomplishments of the NDCU scheme. However, NDCU is only defined for a single ranked list, not supporting utility-based evaluation over multi-session ranked lists. We now describe our new framework, which extends novelty-based evaluation to multi-session retrieval scenarios.

3 New Framework

The core of the new framework is a well-defined probability distribution over user behavior with respect to multiple ranked lists. We define a utility function conditioned on user behavior, and sum over all possible user interactions to obtain an *expectation* of the utility.

Let $l_1, l_2, ..., l_K$ be a sequence of K ranked lists of passages, with lengths given by $|l_1|, |l_2|, ..., |l_K|$, respectively. We define Ω as the space of all possible user browsing patterns – each element $\omega \in \Omega$ denotes a possible way for a user to browse the ranked lists, i.e., to read a specific subset of the passages that appear in the ranked lists. Let P denote a probability distribution over the space Ω, such that $P(w)$ corresponds to how likely it is for a user to read this set of passages. Intuitively, P should assign higher probability to subsets that include passages at top ranks, reflecting common user behavior. We leave the specific details of modeling user behavior to Section 3.1.

Once we have a way of representing different user interaction patterns ω, we can define the utility as a function of ω, i.e. $\mathcal{U}(\omega)$. Note that $\mathcal{U}(\omega)$ is a random quantity, since ω is a random variable. Therefore, the obvious next step is to calculate the expected value of U with respect to the probability distribution defined over Ω. We call this quantity as Expected Global Utility:

$$\text{EGU} = \sum_{\omega \in \Omega} P(\omega)\mathcal{U}(\omega) \tag{4}$$

3.1 User Browsing Patterns

As mentioned earlier, a user can read any subset of the passages presented by the system. We will use Ω to denote the set of all subsets that the user can read. Naturally, the most flexible definition of Ω would be the power set of all passages in the K lists, and the size of such a state space would be $2^{\sum_{i=1}^{K} |l_i|}$. This is a very large state space, leading to difficulties in estimating a probability distribution as well as computing an expectation over the entire space. Another alternative is to restrict the space of possible browsing patterns by assuming that the user browses through each ranked list *top down* without skipping any passage, until he or she decides to stop. Thus, each possible user interaction is now denoted by a K-dimensional vector $\omega = \{s_1, s_2, ..., s_K\}$, such that $s_k \in \{1..|l_k|\}$ denotes the stopping position in the k^{th} ranked list. This leads to a state space of size $\prod_{i=1}^{K} |l_k|$, which is much smaller than the earlier *all-possible-subsets* alternative. We further make a reasonable assumption that the stopping positions in different ranked lists are independent of each other, i.e., $P(\omega) = P(s_1, s_2, ..., s_K) = P(s_1)P(s_2)...P(s_K)$.

The particular form of $P(s)$, i.e., the probability distribution of stopping positions in a ranked list, can be chosen appropriately based on the given domain, user interface, and user behavior. For the purposes of this discussion, we follow Moffat et al. [8] and restrict attention to the geometric distribution with an adjustable (or empirically estimated) parameter, p. However, the standard geometric distribution has an infinite domain, but each ranked list in a distillation system will have a finite length. Therefore, we use a *truncated geometric distribution with a tail mass*, i.e., for a ranked list of length l, the left-over probability mass beyond rank l is assigned to the stopping position l, to reflect the intuition that users who intended to stop before rank l will be oblivious to the limited length of the ranked list, but all users who intended to stop at a rank lower than l will be forced to stop at rank l due to the limited length of the ranked list. Formally the stopping probability distribution for the k^{th} ranked list can be expressed by the following recursive formula:

$$P(S_k = s) = \begin{cases} (1-p)^{s-1}p & 1 \le s < |l_k| \\ 1 - P(S_k < |l_k|) & s = |l_k| \\ 0 & \text{else} \end{cases} \qquad (5)$$

3.2 Utility of Multi-session Ranked Lists Conditioned on User Browsing Patterns

The utility of multi-session ranked lists $l_1, l_2, ..., l_K$ depends on how a user interacts with them. We now define $\mathcal{U}(\omega)$ as the utility of multiple ranked lists conditioned on a user interaction pattern. Recall that $\omega = (s_1, s_2, ..., s_K)$ specifies the stopping positions in each of the ranked lists, allowing us to construct the list of passages actually read by the user for any given ω. We denote this list as $\mathcal{L}(\omega) = \mathcal{L}(s_1, s_2, ..., s_K)$, obtained by concatenating the top $s_1, s_2, ..., s_K$ passages from ranked lists $l_1, l_2, ..., l_K$, respectively. The conditional utility $\mathcal{U}(\omega)$ is defined as:

$$\mathcal{U}(\omega) = \sum_{i=1}^{|\mathcal{L}(\omega)|} G(p_i|\mathcal{L}(\omega)) - aC(p_i) \qquad (6)$$

Comparing this formula with Equation 3, which defines the Discounted Cumulated Utility (DCU) for a single ranked list, we see that utility calculations in the two cases are almost identical, except (i) the single ranked list in DCU is replaced by the synthetic $\mathcal{L}(\omega)$ from the multi-session lists, and (ii) the discounting factor $P(R_i = 1)$ is removed here because each passage in $\mathcal{L}(\omega)$ is assumed to be read by the user.

Substituting $G(.)$ and $C(.)$ in Equation 6 using their definitions from Section 2, we have:

$$\begin{aligned}\mathcal{U}(\omega) &= \sum_{i=1}^{|\mathcal{L}(\omega)|} G(p_i|\mathcal{L}(\omega)) - a \sum_{i=1}^{|\mathcal{L}(\omega)|} C(p_i) \\ &= \sum_{i=1}^{|\mathcal{L}(\omega)|} \sum_{j=1}^{|\Delta|} I(\delta_j, p_i) w(\delta_j, q) \gamma^{n(\delta, \mathcal{L}(\omega), i-1)} - a \sum_{i=1}^{|\mathcal{L}(\omega)|} \mathrm{len}(p_i) \qquad (7)\end{aligned}$$

where Δ is the full set of nuggets in the data collection; $I(\delta_j, p_i) \in \{1, 0\}$ indicates whether or not nugget δ_j is contained in passage p_i, and $\mathrm{len}(p_i)$ is the length of passage p_i.

The first term in Equation 7 is the cumulated gain (CG) from the synthetic list, which can be further calculated as:

$$\begin{aligned}CG(\omega) &= \sum_{j=1}^{|\Delta|} w(\delta_j, q) \left(\sum_{i=1}^{|\mathcal{L}(\omega)|} I(\delta_j, p_i) \gamma^{n(\delta_j, \mathcal{L}(\omega), i-1)} \right) \\ &= \sum_{j=1}^{|\Delta|} w(\delta_j, q) \left(1 + \gamma + \gamma^2 + ... + \gamma^{m(\delta_j, \mathcal{L}(\omega))-1} \right) \\ &= \sum_{j=1}^{|\Delta|} w(\delta_j, q) \frac{1 - \gamma^{m(\delta_j, \mathcal{L}(\omega))}}{1 - \gamma} \qquad (8)\end{aligned}$$

where $m(\delta_j, \mathcal{L}(\omega))$ is the count of passages that contain the nugget. An interesting insight we can obtain from Equation 8 is that the CG value depends on ω only through nugget counts $m(\delta_j, \mathcal{L}(\omega))$ for $j = 1, 2, ..., |\Delta|$. Thus, these nugget counts are the sufficient statistics for calculating CG.

The second term in Equation 7 is the cumulated cost (CC) weighted by a, which is dependent on $\mathcal{L}(\omega)$ only through the count of total word occurrences in the list. Thus the word count is a sufficient statistic for CC, and we denote it by $\mathrm{len}(\mathcal{L}(\omega))$.

Rewriting utility $\mathcal{U}(\omega)$ as a function of the sufficient statistics, we have:

$$\begin{aligned}\mathcal{U}(\omega) &= g(m(\mathcal{L}(\omega)) - a \, \mathrm{len}(\mathcal{L}(\omega)) \qquad (9) \\ &= \frac{1}{1 - \gamma} \sum_{j=1}^{|\Delta|} w(\delta_j, q) \left(1 - \gamma^{m(\delta_j, \mathcal{L}(\omega))} \right) - a \, \mathrm{len}(\mathcal{L}(\omega)) \qquad (10)\end{aligned}$$

Expected Global Utility. Given the utility of multi-session ranked lists conditioned on each specific user browsing pattern, calculation of the expectation over all patterns is straightforward:

$$\mathbb{E}\left[\mathcal{U}(\omega)\right] = \sum_{\omega \in \Omega} P(\omega)\mathcal{U}(\omega)$$
$$= \sum_{s_1=1}^{|l_1|} \dots \sum_{s_K=1}^{|l_K|} \left(\prod_{k=1}^{K} P(s_k) \right) \mathcal{U}(\underbrace{s_1, \dots, s_K}_{\omega}) \tag{11}$$

4 Tractable Computation

Unfortunately, the exact utility calculation quickly becomes computationally intractable as the number and lengths of ranked lists grow. Therefore, we make an approximation. We first rewrite EGU in terms of expected gain and expected cost. Using Equation 9 we have:

$$\mathbb{E}\left[\mathcal{U}(\omega)\right] = \mathbb{E}\left[g(m(\mathcal{L}(\omega)))\right] - a\mathbb{E}\left[\text{len}(\mathcal{L}(\omega))\right] \tag{12}$$

We then approximate the gain[3](the first term above) as:

$$\mathbb{E}\left[g(m(\mathcal{L}(\omega)))\right] \approx g(\mathbb{E}\left[m(\mathcal{L}(\omega))\right]) \tag{13}$$

Thus, instead of calculating the *expected gain* with respect to different browsing patterns, we compute the gain for the *expected browsing patterns* $\mathbb{E}\left[(m(\mathcal{L}(\omega)))\right]$, i.e., the expected number of times each nugget will be read from all the ranked lists.[4]

Since the number of times each nugget will be read in a single ranked list only depends on the possible stopping positions in that list, and is independent of the stopping positions in other ranked lists, the computation can be decomposed into K terms as follows:

$$\mathbb{E}\left[m(\delta_j, \mathcal{L}(\omega))\right] = \sum_{k=1}^{K} \mathbb{E}\left[m(\delta_j, l_k(s_k))\right]$$
$$= \sum_{k=1}^{K} \sum_{s_k=1}^{|l_k|} P(s_k) m(\delta_j, l_k(s_k)) \tag{14}$$

where $m(\delta_j, l_k(s_k))$ denotes the number of times nugget δ_j is read in the k^{th} ranked list when the stopping position is s_k. Thus, the approximate computation requires a sum over $O(|l_1| + |l_2| + \dots + |l_K|)$ terms, instead of the $O(|l_1| \times |l_2| \times$

[3] Cost is easy to calculate due to its simple definition, and does not require any approximation.

[4] We can further approximate gain by moving the expectation operator further inside, i.e. $g(\mathbb{E}\left[m(\mathcal{L}(\omega))\right]) \approx g(m(\mathcal{L}(\mathbb{E}\left[\omega\right])))$, which is equivalent to calculating the gain based on the expected stopping position in each ranked list – in our case – $1/p$, i.e., the expected value of the geometric distribution with parameter p. This corresponds to the approximation used in [7] – a fixed stopping position in each ranked list. However, we do not pursue this extra approximation in the rest of this paper.

... $\times |l_K|$) terms in the original definition, which must consider all combinations of stopping positions in the K ranked lists.

To verify the validity of the approximation, we compared the approximate calculation against the exact `EGU` calculation on randomly generated multi-session ranked lists. The approximate and exact EGU scores were found to be very close to each other.[5]

5 Utility Optimization

An important job of a distillation system is to determine how many passages to select for the user's attention, since reading the system's output requires user effort. However, relevance-based metrics like `MAP` and `NDCG` provide no incentive for the system to produce a limited-length ranked list. On the other hand, `EGU` takes into account the relevance, novelty, and cost of reading, and hence, provides an opportunity to tune the parameters of the distillation system for optimizing its utility.

We consider two ways of adjusting the lengths of the ranked lists: (i) `Fixed length ranked lists`: Using a held-off (validation) dataset, the optimal length of ranked lists (e.g., 5, 10, 20, or 50 passages) is determined, and then held fixed for the test phase, and (ii) `Variable length ranked lists`: Instead of fixing the absolute length of the ranked lists, the relevance and novelty thresholds of the system are tuned and then these thresholds are held fixed for the test phase. Only the passages whose relevance and novelty scores are both above the corresponding thresholds remain in the ranked lists. The second approach is more flexible since it allows the system to account for varying amounts of relevant and novel information in each retrieval session by adjusting the length of its ranked lists accordingly.

6 Experiments

To demonstrate the effectiveness of the proposed framework for evaluating and optimizing the utility of distillation systems, we conducted controlled experiments with a state-of-the-art distillation system on a benchmark corpus.

Dataset. `TDT4` was a benchmark corpus used in Topic Detection and Tracking (`TDT2002` and `TDT2003`) evaluations. It consists of over 90,000 articles from various news sources published between October 2000 and January 2001. This corpus was extended for distillation evaluations by identifying 12 actionable events and defining information distillation tasks on them, as described in [5,12]. Following [12], we divided the 4-month span of the corpus into 10 chunks, each comprising 12 consecutive days. A distillation system is expected to produce a ranked list of documents at the end of each chunk, receive feedback from the user, and then produce a new ranked list for the next chunk, and so on. We split the data into a

[5] See detailed results at `http://nyc.lti.cs.cmu.edu/papers/utility/`

validation set and a test set, each consisting of 6 events corresponding to 59 and 45 queries, respectively. We use the validation set to tune the lengths of ranked lists in the two ways mentioned in Section 5, and evaluate the performance of the system on the test set.

Metrics. We measure performance using two metrics: (i) Mean Average Precision (MAP) [3], which is a popular metric used for relevance-based evaluation, and (ii) the proposed metric EGU with parameters $a = 0.01$ and $\gamma = 0.1$ (see Section 3.2).

Systems. We use the CMU Adaptive Filtering Engine (CAFÉ) [12], which is a state-of-the-art distillation system that combines adaptive filtering, ranked retrieval, and novelty detection. To understand the behavior of the proposed metric, we run the system in different configurations by using various combinations of ranked retrieval (IR), adaptive filtering with feedback[6] (AF), and novelty detection (ND).

To assess the base performance of CAFÉ, we compared its Mean Average Precision (MAP) in pure retrieval mode (i.e., CAFÉ:IR) with that of Indri [10], which is a state-of-the-art retrieval engine. The MAP scores were found to be as follows: CAFÉ:IR – 0.3677, and Indri – 0.3798, which validates CAFÉ as a relatively strong system, given that Indri is a well-established retrieval engine.

Ranked list length optimization. As described in Section 5, we try two variants of controlling the lengths of ranked lists – Fixed and Variable.

Table 1. EGU and MAP scores for various configurations of CAFÉ

System configuration	Ranked list lengths	EGU	MAP
CAFÉ:IR	Fixed	0.2592	0.3677
CAFÉ:IR+ND	Fixed	0.2640	0.3534
CAFÉ:IR+AF	Fixed	0.3001	**0.5019**
CAFÉ:IR+AF+ND	Fixed	0.3014	0.4737
CAFÉ:IR+AF	Variable	0.3101	**0.5019**
CAFÉ:IR+AF+ND	Variable	**0.4701**	0.4737

6.1 Main Results

Table 1 shows the MAP and EGU scores obtained by running CAFÉ with various combinations of three techniques: ranked retrieval, adaptive filtering and novelty

[6] In our experiments, we have followed the standard adaptive filtering convention of providing user feedback on all passages that are presented by the system. If user feedback is to be modeled as stochastic, then our feedback-based results represent the upper bound on performance since the system would receive feedback on only a subset of passages.

detection, and also with two methods for optimizing the ranked list lengths: fixed lengths, and variable lengths.

The addition of feedback through adaptive filtering (AF) helps the system in finding more relevant passages in subsequent sessions. This manifests as improved performance in terms of both EGU as well as MAP, showing that both metrics are sensitive to the relevance of presented passages.

Now consider the effect of enabling novelty detection (ND), which directs the system to detect and remove redundant passages from each ranked list. Keeping other settings constant, we see that novelty detection improves the EGU scores, while decreasing the MAP scores, since MAP blindly favors the presence of relevant passages even if some of them are redundant, and hence, useless to the user.

Next, we focus on the two strategies for optimizing the ranked list lengths for each query. As described in Section 5, the Fixed mode retrieves a pre-determined number of passages for each query in each chunk, even if that chunk does not have any relevant and novel information for the given query. This problem is solved by the Variable mode, which indirectly controls the lengths of ranked lists by optimizing its utility (relevance and novelty) thresholds. The EGU scores show better performance when Variable mode is used. However, the MAP scores do not change between the Fixed and Variable mode, because in both cases, MAP will favor the longest possible ranked lists. In other words, MAP scores can never be improved by reducing the length of the ranked list, since MAP is oblivious to the reading cost associated with the presented information.

The best performance in terms of EGU is obtained when novelty detection is combined with variable ranked list lengths based on optimal relevance and novelty thresholds (CAFÉ:IR+AF+ND-Variable). We obtain an EGU score of 0.4701, a 51.6% improvement over not using novelty detection (CAFÉ:IR+AF-Variable), a 56% improvement over not using variable length ranked lists (CAFÉ:IR+AF+ND-Fixed), and a 56.6% improvement over no novelty detection and no variable length lists (CAFÉ:IR+AF-Fixed). Interestingly, neither novelty detection nor variable ranked list length mode can alone boost the performance significantly.

7 Concluding Remarks

In this paper, we have proposed the first theoretical framework where non-deterministic user interactions over multi-session ranked lists are taken into account for evaluating and optimizing the utility of distillation systems. This model flexibility comes at the cost of increased computational complexity, which we address using an efficient approximation technique. We conducted the first utility-based evaluation over multiple-session search scenarios defined on the TDT4 corpus of news stories, and show that a distillation system can obtain significant (56.6%) utility enhancement by combining multi-session adaptive filtering with novelty detection and utility-based optimization of system parameters for optimal ranked list lengths. Our framework can naturally accommodate more sophisticated probabilistic models of user behavior that go beyond the geometric distribution over stopping positions as described in this paper. For instance, an interesting direction would be to model the stopping probability as dependent on user satisfaction.

Acknowledgements. This work is supported by the National Science Foundation (NSF) under grant IIS_0704689. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

References

1. Allan, J., Wade, C., Bolivar, A.: Retrieval and novelty detection at the sentence level. In: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, pp. 314–321 (2003)
2. Babko-Malaya, O.: Annotation of Nuggets and Relevance in GALE Distillation Evaluation. In: Proceedings LREC 2008 (2008)
3. Buckley, C., Voorhees, E.M.: Retrieval system evaluation. TREC: Experiment and Evaluation in Information Retrieval, 53–75 (2005)
4. Clarke, C.L.A., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkan, A., Büttcher, S., MacKinnon, I.: Novelty and diversity in information retrieval evaluation. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 659–666 (2008)
5. He, D., Brusilovsky, P., Ahn, J., Grady, J., Farzan, R., Peng, Y., Yang, Y., Rogati, M.: An evaluation of adaptive filtering in the context of realistic task-based information exploration. In: Information Processing and Management (2008)
6. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems (TOIS) (4), 422–446 (2002)
7. Järvelin, K., Price, S., Delcambre, L., Nielsen, M.L.: Discounted cumulated gain based evaluation of multiple-query IR sessions. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 4–15. Springer, Heidelberg (2008)
8. Moffat, A., Zobel, J.: Rank-Biased Precision for Measurement of Retrieval Effectiveness. ACM Transactions on Information Systems, 1–27 (2008)
9. Robertson, S.: A new interpretation of average precision. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 689–690 (2008)
10. Strohman, T., Metzler, D., Turtle, H., Croft, W.B.: Indri: A language model-based serach engine for complex queries. In: Proceedings of the International Conference on Intelligence Analysis (2004)
11. White, J.V., Hunter, D., Goldstein, J.D.: Statistical Evaluation of Information Distillation Systems. In: Proceedings of the Sixth International Language Resources and Evaluation LREC, vol. 8
12. Yang, Y., Lad, A., Lao, N., Harpale, A., Kisiel, B., Rogati, M.: Utility-based information distillation over temporally sequenced documents. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 31–38 (2007)

Specificity Aboutness in XML Retrieval

Tobias Blanke and Mounia Lalmas

Department of Computing Science, University of Glasgow
tobias.blanke@dcs.gla.ac.uk, mounia@acm.org

Abstract. This paper presents a theoretical methodology to evaluate filters in XML retrieval. Theoretical evaluation is concerned with the formal investigation of qualitative properties of retrieval models. XML retrieval deals with retrieving those document components that specifically answer a query, and filters are a method of delivering the most focused answers. Our theoretical evaluation will critically analyse how filters achieve this.

1 Introduction

According to INEX, the evaluation initiative for XML retrieval [6], the aim of XML retrieval is to retrieve not only relevant document components, but those at the right level of granularity, i.e. those that specifically answer a query. To evaluate how effective XML retrieval approaches are, it is necessary to consider whether the 'right' level of the structure is correctly identified. For this purpose, INEX has developed a new relevance criterium next to general relevance, which measures how focused an XML element is with respect to an information need. The general relevance of an element is captured in the INEX exhaustivity dimension[1] while the specificity dimension indicates the focus.

In this paper, we analyze retrieval models developed at INEX that aimed at delivering results that specifically answer a query. Delivering these so-called *most specific* answers has proven to be a complex retrieval task. In addition, it has been noted that traditional information retrieval (IR) evaluation might not be sufficient to properly assess the effectiveness of such more complex retrieval tasks [10]. This paper proposes an alternative theoretical evaluation that complements an experimental evaluation, especially when dealing with complex retrieval tasks such as those developed for XML retrieval.

A theoretical evaluation can be done through the use of a meta-theory, as proposed in previous work based on the logical approach to IR [7]. Van Rijsbergen and others have expressed logical relevance in terms of the implication $d \rightarrow q$ [10]. Chiaramella [4] used two implications to describe the XML retrieval task[2]: $d \rightarrow q$ characterizing exhaustivity and $q \rightarrow d$ characterizing specificity.

[1] Since 2006, INEX does not refer to exhaustivity anymore, just relevance and specificity.

[2] When this work was published, it referred to the more general case of structured document retrieval, for which XML retrieval is a special case.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 176–187, 2009.

Following Huibers' work [7], we call such implications between query and document *aboutness*. IR models propose specific ways to implement the aboutness of a document to a query. With this view in mind, the theoretical evaluation of an IR model thus consists of characterizing aboutness and investigating its underlying reasoning process.

Aboutness has been discussed sporadically in IR literature, most notably in the work of Lalmas and Van Rijsbergen [11], Huibers and Bruza [3], and recently Wong et al.[13]. However, aboutness has yet to be applied to more complex IR tasks such as those occurring in XML retrieval. In this paper, we use the concept of aboutness to evaluate XML retrieval models that aimed at identifying the most specific document components for a query.

In INEX, the retrieval task that aims at finding the most specific answers has been referred to as the focused task. This is to be compared to the thorough task, that aims at estimating the relevance of document components to a query. In this latter task, all relevant document components are to be identified, and then ranked according to their degree of relevance. In the focused task, the result set should consist of non-overlapping document components, ranked according to how specific they are to the query. Overlap occurs when a document component (e.g. a section) and one of its descendent (e.g. a paragraph in this section) or ascendent (e.g. the chapter containing that section) are both returned as answers. The aim of the focused task is therefore to identify among overlapping document components, the component that is the most specific to the query, and to return it as what is referred to as a *focused* answer. In this paper, we concentrate on retrieval models developed for the focused task at INEX 2005, as the fundamentals of these models have not changed much since. In addition, we restrict ourselves to models aiming at delivering these focused answers for content-only queries.

Models developed at INEX to implement the focused retrieval task can be viewed as filters. Indeed, these models mostly consist of the post-processing of an answer set produced by models aiming at implementing the thorough retrieval task. The post-processing phase consists of eliminating all but the most focused document components from the answer set. We therefore analyze filters as an aboutness decision in their own right. Several types of filters have been developed in INEX. The most popular filter is the so-called brute-force one, which eliminates all but the most relevant elements[3] on a particular XML path. However, in the experimental evaluation, it performs less well than others that look at the relationships between elements [8]. We therefore compare it to an alternative approach based on the re-ranking of elements.

This paper is organised as follows: In Section 2, we introduce the background of our theoretical evaluation methodology. In Section 3, we briefly draw on earlier results to demonstrate parts of the theoretical evaluation of two XML retrieval approaches implementing the thorough task. We then introduce in Section 4 our theoretical methodology to analyse filters as aboutness decisions, before applying it to the brute-force and re-ranking filtering models in Section 5. Finally, we relate our findings to those of the experimental evaluation in INEX in Section 6.

[3] In this paper, elements and document components are used interchangeably.

2 Theoretical Evaluation Background

In this section, we introduce the steps of our theoretical evaluation methodology (see [2] for a complete overview). A theoretical evaluation methodology needs a formalism powerful enough to characterize the fundamental properties of retrieval models. Following Huibers, we use Situation Theory (ST), developed by Barwise and Perry [1], for this purpose. ST is a mathematical theory of meaning and information with *situations* as primitives [7]. Situations are partial descriptions of the world and are composed of *infons*. For IR modelling, queries and documents are modelled as situations, while infons represent a model's information items like keywords or phrases.

Using ST, we model documents and queries as situations [3]. Let document D and query Q be situations, then $D \Box\!\!\leadsto Q$ means that the information in D is about the information need expressed in Q. For instance, in standard IR, a document containing 'garden' and 'house' would be about a query asking for 'garden'. Likewise, $D \Box\!\!\not\leadsto Q$ symbolises that D is not about Q. For XML retrieval, we can use Chiaramella's distinction and say that $D \Box\!\!\leadsto Q$ symbolizes exhaustivity and $Q \Box\!\!\leadsto D$ specificity. With $\otimes$, we formalise the composition of situations, while $\equiv$ states that two situations are equivalent, i.e. they contain the same information.

Translation is the symbolic representation of an IR model's handling of information using a formal language. It is formally represented by a function *map* that 'maps' situations to their formal representation. In IR, mapping a document (or a document component) to its formal representation corresponds to the indexing process. For standard IR, the outcome would consist of a set of infons represented by $\langle\langle k \rangle\rangle$, where k stands for an indexing term. A set of infons is a situation: $\{\langle\langle k_1 \rangle\rangle, \langle\langle k_2 \rangle\rangle\}$. An example would be $\{\langle\langle house \rangle\rangle, \langle\langle garden \rangle\rangle\}$.

For representing information in XML retrieval, we furthermore use N-ary relationships R between infons i_j, to model relationships: $\langle\langle R, i_1, ..., i_n \rangle\rangle$. For instance, a section with two paragraphs will be symbolized by: $\{\langle\langle ElementType, Sec, s \rangle\rangle, \langle\langle ElementType, Para, p_1 \rangle\rangle, \langle\langle Value, garden, p_1 \rangle\rangle, \langle\langle ElementType, Para, p_2 \rangle\rangle, \langle\langle Value, house, p_2 \rangle\rangle, \langle\langle Parent, s, p_1 \rangle\rangle, \langle\langle Parent, s, p_2 \rangle\rangle\}$. This reflects the fact that each XML element has an element type infon, expressed with the relation *ElementType*. Content infons (i.e. the actual text in the element) are modeled as $Values$. The relation *Parent* expresses that the two paragraphs (p_1 and p_2) are the children of the section (s). Translation is therefore based on building a document representation through indexing, which according to Van Rijsbergen [12] leads to the view that index terms represent properties of documents (or document components), which may then be studied.

Next to the symbolic characterization of an IR model, we need means to describe the functional behavior of the model, i.e. what makes a document (or document component in XML retrieval) about a query. This is done through so-called *reasoning rules*. Indeed, an IR model's aboutness decision is specified by the reasoning rules it incorporates. These can be either fully, partially, or not at all supported. Together with the symbolic representation of documents, queries and information for an IR model, they make up the *aboutness decision* *system* that characterizes how the model decides that a document (or document component) is relevant to query.

An example of a rule is Left Monotonic Union (LMU), which plays an important role in our theoretical study of filters. LMU states that if a document D is about a query Q, then also the composition of D and D':[4]

- LMU: If $D \Box\!\!\!\rightsquigarrow Q$, then also $D \otimes D' \Box\!\!\!\rightsquigarrow Q$.

By comparing the reasoning rules each decision system incorporates and the way it does so, we are able to give an overall comparison of the behaviour of the retrieval model characterized by the aboutness decision system.

There are over 20 reasoning rules to be considered in the theoretical analysis of retrieval models (see [7] and [13]). In this paper, we restrict ourselves to the following rules (including LMU), as they are sufficient for our investigation:

- Reflexivity: $S \Box\!\!\!\rightsquigarrow S$.
- Transitivity: If $S \Box\!\!\!\rightsquigarrow T$ and $T \Box\!\!\!\rightsquigarrow U$, then also $S \Box\!\!\!\rightsquigarrow U$.
- Euclid: If $S \Box\!\!\!\rightsquigarrow T$ and $S \Box\!\!\!\rightsquigarrow U$, then also $T \Box\!\!\!\rightsquigarrow U$.
- Mix: If $S \Box\!\!\!\rightsquigarrow T$ and $U \Box\!\!\!\rightsquigarrow T$, then also $S \otimes U \Box\!\!\!\rightsquigarrow T$.

3 Aboutness in XML Retrieval

To carry out our theoretical evaluation of filters (the focused retrieval task at INEX), it is necessary to present (albeit briefly) the theoretical evaluation of models developed for the thorough task. This is because of the relationships between the tasks, one being a post-processing of the other. This also provides an illustration of our theoretical methodology. We focus on two such models, both building upon well-known flat document retrieval models.

3.1 Vector Space Model

A vector space model for XML retrieval is presented in [8]. There, XML documents are split into several disjoint indexes of the most useful components (which can be determined for a given application). In the model, a standard vector space approach is used to retrieve from a query (Q) XML elements (D) instead of full documents:

$$rsv(Q,D) = \frac{\sum_{t_i \in \{Q \cap D\}} w_Q(t_i) * w_D(t_i) * idf(t_i)}{\|Q\| * \|D\|}$$

where $w_Q(t) = \frac{log(TF_Q(t))}{log(AvgTF_Q)}$ and $w_D(t) = \frac{log(TF_D(t))}{log(AvgTF_D)}$. $||Q||$ and $||D||$ are the numbers of unique terms in Q and D, respectively. Both are scaled by the average document length in the collection [8].[5]

[4] Throughout the paper, we use upper case letters from the middle of alphabet such as S, T, for situations if we are not talking about queries and document components. In that case we use Q and D. Anything that situations are made of, e.g. keywords but also structural relationships, is symbolized with letters from the beginning of the alphabet like A or B.

[5] The obtained scores are modified using an Automatic Query Refinement (AQR) approach based on Lexical Affinity (LA), which is out of scope for this paper.

Structure is used in the model mainly to allocate document components across different indexes. The translation is limited to those infons of document components most commonly assessed as relevant. In the INEX 2005 collection, these included paragraph ('Para','Para1'), subsection ('SS1', 'SS2'), section ('Sec'), etc. Regarding the translation, let A by a document component and e be an element type, then $map(A) = \{\langle\langle ElementType, e, i\rangle\rangle, \langle\langle Value, t, i\rangle\rangle | e \in \{Art, Abs, Sec, SS1, SS2, Para, Para1\}$.

For the aboutness decision, let Q be a query and D be a document component. That we consider components instead of full documents is the main difference to the flat vector space model. The *XML retrieval vector space aboutness decision* is then defined by:

$$D \,\Box\!\!\rightsquigarrow Q \text{ if and only if } rsv(D, Q) \geq n$$

In the model, only the top N documents are considered. We call the value that has to be reached in order to be part of the top N documents n. Thus, the model implements thresholded vector space retrieval [13].

Regarding the reasoning rules, we can prove that LMU is conditionally satisfied. As the model implements thresholded vector space retrieval, the extension of D to $D \otimes D'$ will only continue to be about Q if there is still sufficient information overlap between $D \otimes D'$ and Q (the full proof can be found in [2]).

3.2 Language Models

A second model, that was based on a model for flat document retrieval and that performed well at INEX, uses language modeling [9]. The model builds several indexes, each of which is separately populated: one for all elements, one length based one, one for elements frequently assessed as relevant, one for sections. The full article is kept in another index. A language model for each document component is calculated by interpolating the element ($P_{mle}(t_i|e)$), the document ($P_{mle}(t_i|d)$) and the collection ($P_{mle}(t_i)$) language models:

$$P(t_i|e) = \lambda_e * P_{mle}(t_i|e) + \lambda_d * P_{mle}(t_i|d) + (1 - \lambda_e - \lambda_d) * P_{mle}(t_i)$$

This model is built on the decision that a document component D is about a query Q if and only if the information in Q can be found in the indexes. We actually have different aboutness decisions, depending on which index is used to generate the element language model. We therefore must provide a *map* function to translate infons for each chosen index. We only demonstrate 2 of the 6 indexes, as the others are built very similarly. Let A be an element with term t and an element type e:

- Length based index: $map_{length}(A) \equiv \{\langle\langle ElementType, e, i\rangle\rangle, \langle\langle Value, t, i\rangle\rangle | |A| > \kappa\}$
- Section index: $map_{sec}(A) \equiv \{\langle\langle ElementType, e, i\rangle\rangle, \langle\langle Value, t, i\rangle\rangle | e \in \{Sec\}\}$

where κ is a threshold that discards small elements. Apart from the article index, the main difference to a flat document language model is the division into document components instead of documents. This *XML language model retrieval aboutness decision* is the same as for the flat document language model: D about Q if and if only $P(t_i|e) > \theta$. The threshold θ is the smoothing value, which is the collection language model $(1 - \lambda_e - \lambda_d) * P_{mle}(t_i)$. Contrary to the vector space model threshold, it is internal to the aboutness decision, as it is dependent on the overall distribution of the terms in the collection. This allows the model to be adjusted well to specific collections like INEX. We have shown in [2] that LMU is unconditionally supported.

Both discussed models performed well at INEX, which we could relate to their aboutness behaviour in [2]. However, both models performed better for the thorough retrieval task than for the task aiming at returning the most focused elements, i.e. the focused retrieval task. This paper provides a theoretical explanation for this behaviour.

4 Defining Specificity Aboutness

In this section, we describe our theoretical methodology to evaluate filters. We rely on some initial work by Huibers on the relationship between the filter aboutness system (characterizing the focused task) and the corresponding underlying aboutness system (characterizing the thorough task) [7], which we adapt to the requirements of XML retrieval. We go beyond his work by actually applying his theoretical work to analyze two filters developed at INEX in Section 5.

As already explained, the task of finding the most focused elements consists of filtering the ranked result list produced by an XML retrieval model like the two described in Section 3. Generating this ranked result list is itself based on an aboutness decision system, which characterizes the model used to deliver that list. Thus, with filtering, a further aboutness decision is applied, one which removes overlapping elements from the result list.[6]

Huibers [7] describes that one aboutness decision system is a filter to another aboutness decision system if the two corresponding aboutness systems are embedded — meaning their reasoning behaviour is related by supporting the same or sufficiently similar properties. In the context of XML retrieval, this translates to having to relate the aboutness decision system associated with the model for the focused task to that of the underlying aboutness system associated with the model used to generate the ranked list (the thorough task) to then be filtered.

The theoretical analysis of filter is done in three steps. We first formalize the translation process, as we did in Section 3 for retrieval systems. Secondly, we

[6] It should be pointed out that the use of filters is not exclusive to XML retrieval. Filters are used in IR to improve performance [7], if, for instance, at first a fast but less accurate approach is used to identify relevant documents from a very large set documents, and then a second retrieval system is used to search the initial result set more accurately. Pseudo-relevance feedback and passage retrieval are examples of such a process.

identify the reasoning rules associated with the filter. Finally, we analyse the relationship between the filter and the underlying aboutness systems. For the later, we make use of the filtering function f-*answer* defined in [7], which we adapt to XML retrieval:

Definition 1. *Let A_p, B_p be aboutness systems and $\mathcal{D}$ be a set of documents and Q be a query. The filtering function f-answer of A_p with respect to B_p is defined by: f-answer$(A_p; B_p; Q; \mathcal{D}) = answer(A_p; Q; answer(B_p; Q; \mathcal{D}))$, where answer describes a function that delivers an answer set from the set $\mathcal{D}$ based on query Q.*

Using this definition, we can investigate the filtering process by looking at the relationship between f-*answer* and *answer*. Without going into detail, Huibers has identified three important distinctions between f-*answer* and *answer* [7]:

- A filtering function f-$answer(A_p; B_p; Q; \mathcal{D})$ is called *useless* if for all sets of documents $\mathcal{D}$ and queries Q f-$answer(A_p; B_p; Q; \mathcal{D}) = answer(B_p; Q; \mathcal{D})$. An example of a useless filter is the application of the coordinate retrieval model as a filter to an answer set generated by simple vector space retrieval, as both are based on the same aboutness decisions, according to which a document D is about a query Q if both share information items.
- The aboutness systems A_p and B_p are said to be *f-equivalent* if and only if f-$answer(A_p; B_p; Q; \mathcal{D}) = answer(A_p; Q; \mathcal{D})$. An example of an *f-equivalent* filter is to use strict coordinate retrieval to filter a result set generated by vector space retrieval. Strict coordinate retrieval defines that a document D is about a query Q if and only if the information items of Q are a subset of the information items in D. This delivers a subset of the answer set from simple vector space retrieval, for which D is about Q if they share information. Strict coordinate retrieval therefore fully determines the final answer set.
- A_p and B_p are said to *intersect* if and only if the filter is neither useless nor f-*equivalent*.

In our analysis of the relationship between f-*answer* and *answer*, we first determine whether a filter is 'useless', i.e. the filtering function does not change the original answer set. If this is not the case, next we investigate whether the filter f-*answer* uses f-*equivalent* aboutness systems. We call a filter aboutness system to be f-*equivalent*, if its A_p alone will determine the final result set. If the filter is not useless and not f-*equivalent* with regard to the underlying aboutness system, we then define how the filter and underlying aboutness system 'intersect' by comparing their aboutness properties.

The following section will demonstrate the presented methodology for the analysis of two filers at INEX.

5 Applying Specificity Aboutness at INEX

Two main types of models have been proposed for the focused task at INEX: a simple model that keeps the highest ranked element of each XML path and a more complex model that takes into account the relations in the tree hierarchy between retrieved elements.

5.1 Brute-Force Filter

Our first method of removing overlap in the result set of an XML retrieval model has also been referred to as 'brute-force filter', because only the highest scored element from each of the paths is selected. The advantage of this filter is that it is relatively easy to implement and that it can be used on top of any kind of underlying aboutness system.

Aboutness decision. The aboutness decision of brute-force filtering can be defined as:

$$D \text{ about } Q \text{ if and only if } rsv(D,Q) = max(rsv_u(D,Q))$$

$max(rsv_u(Q,D))$ is delivering the XML element with the maximum retrieval status value for the underlying aboutness system. For the translation, let A be a document component, e_n element types, k_n values in an element, and i an identifier to enumerate all $\{1,...,n\}$ elements in an XML tree in a depth-first traversal manner:

$map(A) = \{\langle\langle ElementType, e_1, i_1\rangle\rangle, \langle\langle ElementType, e_2, i_2\rangle\rangle, \langle\langle Parent, i_1, i_2\rangle\rangle,$ $..., \langle\langle ElementType, e_n, i_n\rangle\rangle, \langle\langle Parent, i_{n-1}, i_n\rangle\rangle, \langle\langle Value, e_n, k_1\rangle\rangle, ...,\langle\langle Value, e_n,$ $k_n\rangle\rangle\} | \forall i_i \in \{\langle\langle Parent, i_i, i_k\rangle\rangle\}, count(i_i) = 1\}.$

The translation expresses that we only consider elements on the same XPath, meaning each element is the parent and the child of exactly one other element, unless it is the root or leaf element.

Reasoning behaviour. We now continue analysing the functional behaviour of brute-force filtering using the reasoning rules from Section 2. Reflexivity holds for brute-force filtering. A maximum element will be about itself. More interesting are those reasoning rules that are not supported: The Transitivity rule, for instance, is not supported, as two situations cannot be the maximum scoring answers towards the same query. If T is the maximum scoring answer to U, S cannot be the maximum scoring answer to U, too. This means whatever the status of Transitivity in an aboutness system, if we apply brute-force filtering on top of it, it will not be supported. The same applies for Euclid from Section 2: If S is the maximum scoring answer to U, how could T be the maximum scoring answer to the same U, too? This means Euclid is never supported.

Mix is another rule that cannot be supported. It states that with the assumptions $S \,\Box\!\!\!\rightsquigarrow U$ and $T \,\Box\!\!\!\rightsquigarrow U$, we can also say that $S \otimes T \,\Box\!\!\!\rightsquigarrow U$. S and T, however, cannot be at the same time the maximum answer to U. The assumptions contradict each other. LMU would imply in the context of brute-force filtering that if one extends S to $S \otimes U$ and aboutness would be preserved for both, both S and $S \otimes U$ would be maximum scoring answers, which is a contradiction. This means LMU is not supported either.

All the rules analysed in this section are important in the analysis of XML retrieval models' behaviour [2]. When we analyse the experimental results related to brute-force filtering in Section 6, we shall see the impact of excluding the rules' reasoning behaviour.

F-answer. In this section, we shall look at the relation between f-*answer* and *answer*. First, we need to show that the brute-force filter is not *useless*. This can be formally proven by demonstrating that the aboutness systems of filter and underlying system differ in at least one reasoning characteristics — be it a certain rule, be it a single condition of this rule. We have just seen that brute-force filtering disallows LMU, Transitivity, etc., which means it is not useless as a filter for both the XML vector space and language model retrieval models (and many XML retrieval approaches we have analysed in [2]). As $max(rsv_u(D,Q))$ is dependent on the underlying retrieval status value rsv_u, brute-force filtering is also not f-*equivalent*.

As the filter is neither useless nor f-*equivalent*, neither brute-force filtering nor the underlying aboutness systems from Section 3 fully determine the outcome of combining both. They 'intersect', which means that we need to look at the differences in reasoning behaviour, the filter creates: E.g., LMU reasoning is excluded, which will change any aboutness system that follows the strict structural constraints of XML documents: If an element is a child, it will share information with its parent. This means for language modeling from Section 3.2, for instance, that both are about the same queries. However, such aboutness due to overlap in (redundant) information is what is supposed to be excluded by brute-force filtering.

We analyse the impact of brute-force filtering on the experimental results in INEX 2005 in Section 6, but first we look at a second alternative approach to dealing with overlapping elements: re-ranking. The assumption is here that sometimes overlap can be beneficial. In [8], they used a similar kind of re-ranking and found that it delivers better performance than the brute-force filtering alone.

5.2 Controlling the Overlap: Re-ranking Approach

The next approach [5] we present will re-rank the elements with a new context-dependent retrieval status value, but not entirely eliminate overlapping elements. The approach is based on iteratively reducing the score of those elements that contain highly relevant elements. The input into the re-ranking method is a list of XML elements x. These are each associated with $x.\overrightarrow{f}$ as the term frequency vector per query term and with $x.\overrightarrow{g}$ as the adjustment vector, and other information required to process the algorithm such as the set of children per element. The adjustment of each term x_t is based on $x_t = f_t - \alpha * g_t$, where α is an adjustment weight. For parents y containing a highly scoring child x their adjustment score $y.\overrightarrow{g}$ will be increased. For the children of highly ranked parents, we know that its terms have already been considered in the reported parent element. Hence, its $x.\overrightarrow{g}$ will become $y.\overrightarrow{f}$. The tree is traversed until all elements are covered and re-ranked.

Aboutness decision. According to the algorithm in [5], no element will be filtered out unless the adjusted score becomes 0. Therefore, the aboutness decision is described by:

$$D \text{ about } Q \text{ if and only if } rsv_{adjusted}(D,Q) > 0$$

The translation of the model is out of scope for this paper, as it would require a deeper analysis of how XML structures can be translated into situations. We have done that analysis in [2].

Reasoning behaviour. The first reasoning property to look at will be Reflexivity, which as seen in Section 2 states that $S \Box\!\!\rightsquigarrow S$. Reflexivity is not given. With $S \Box\!\!\rightsquigarrow S$, then $f_t = g_t$. If in $x_t = f_t - \alpha * g_t$, $\alpha = 1$ [5], then $x_t = f_t - 1 * g_t$, which means $rsv_{adjusted} = 0$, with $f_t = g_t$. Thus, Reflexivity is not supported.

Re-ranking does not fundamentally change the aboutness decision of the XML retrieval models but adds emphasis to the ranking of elements. For our analysis of the impact of filters we therefore need to relate it to the models we have developed in Section 3 directly. For both models, Transitivity, Euclid and Mix behaviour, will not be changed. LMU would be given if $S \otimes U \Box\!\!\rightsquigarrow T$ and $S \Box\!\!\rightsquigarrow T$ are given. Regarding the XML vector space model, re-ranking with $x_t = f_t - \alpha * g_t$ can of course reduce the extension to fall below the threshold n. This will mainly effect the children of the highly ranked parents. LMU is only conditionally supported if re-ranking does not lower the retrieval result to fall below n. An interesting case forms the language modeling approach in Section 3.2. Its internal threshold based on the smoothing value might be missed if the added information leads to a re-ranking below the smoothing value. Therefore, applying re-ranking on top of language modeling means that LMU is now conditionally supported, while language modeling alone fully supported LMU.

F-answer. Re-ranking is certainly not *useless*, because the LMU thresholds for vector space retrieval and language modeling have been changed. It will not be f-*equivalent* either, as it is dependent on the underlying aboutness decision, because re-ranking is a function of the original retrieval status value. Thus, re-ranking will also be ‘intersecting’. Reflexivity is changed through the impact of α. That Transitivity and Mix behaviour is preserved is a clear advantage towards the brute-force filtering approach, as both are important properties of XML retrieval behaviour [2]. In particular the support for Mix, will add to the better performance of the model in the experimental results.

In the next section, we briefly look at how conclusions from the theoretical evaluation of both filters help explain experimental behaviour in INEX 2005.

6 Impact of Filters on Experimental Behaviour at INEX

We first investigate the impact of brute-force filtering on the experimental behaviour in INEX 2005. The XML vector space retrieval model has been overall very successful in the experimental evaluation in INEX 2005 [8], but its performance decreases for the tasks to deliver only non-overlapping document components, ranked according to how specific they are to the query. They implemented these tasks by using various filters. Particularly, in their run which used simple brute-force filtering, the performance was much worse. The situation is similar for the XML retrieval model based on language modelling [9]. Its performance decreases, too, when brute-force filtering is used to filter the original language modeling retrieval results.

If we try to understand why brute-force filtering decreases performance in XML retrieval, two changes of reasoning properties are highly conclusive:

1. LMU is not supported by brute-force filtering. The XML vector space retrieval model, for instance, successfully used conditions on LMU reasoning to adjust the behaviour of flat document vector space retrieval to the requirements of XML retrieval [2]. This ability is lost once the brute-force filter is applied, which will explain a decrease in performance.
2. Mix is not supported by brute-force filtering. Among other things, Mix describes that, if two children D and D' are about a query, then their parent item $D \otimes D'$ will also be about the same query. This behaviour is typical to XML based resasoning. If it is not supported, problems might arise, such as the elimination of potentially highly relevant children. Say, we have one relevant child and a more relevant parent, then the child will be eliminated from the result set after applying brute-force filtering. Another child of the same parent that is about the same query, will also be eliminated, as the parent is already chosen for its path. However, this child might be highly relevant, too.

Looking at the second filter, re-ranking, it is difficult to make general statements regarding its impact on XML retrieval, as it has been developed for a particular model [5]. The authors, however, report limitations of their algorithm according to their experimental evaluation [5]. From a theoretical evaluation point of view, an immediate recommendation on how to potentially improve the model would be to introduce a threshold to control the monotonic behaviour of the re-ranking aboutness decision: Only if $rsv_{adjusted}(D, Q) > \theta$, the element would be reported. We have seen in earlier theoretical evaluations [2], that thresholds effectively add to the control of the monotonic behaviour and improve models' performance.

7 Conclusion

In this paper, we have shown how a theoretical evaluation (in this paper based on ST), can aid the analysis of filters in XML retrieval. To this end, we introduced a theoretical evaluation methodology to help investigate filters based on an ST formalism. We have considered filters as a second layer aboutness decision and asked how they influence the underlying aboutness system. We could do so, as we regarded them as aboutness systems. This has led to conclusions about why and how they change the performance of their underlying systems in the experimental evaluation in INEX. Our primary interest has been whether the filters are suitable extensions of the underlying aboutness decision. We could show how particularly brute-force filters significantly change the underlying aboutness behaviour of the retrieval models. In the future, we we would like to deepen our analysis of how filters for focussed retrieval have an impact on aboutness behaviour, especially on specificity aboutness, by analysing in more detail the impact on experimental results in INEX 2005.

Acknowledgements. Mounia Lalmas is currently funded by Microsoft Research/Royal Academy of Engineering.

References

1. Barwise, J., Perry, J.: Situations and Attitudes. MIT Press, Cambridge (1983)
2. Blanke, T., Lalmas, M.: A framework for the theoretical evaluation of XML retrieval. Paper in preparation for publication
3. Bruza, P.D., Huibers, T.W.C.: Investigating aboutness axioms using information fields. In: ACM SIGIR 1994, pp. 112–121 (1994)
4. Chiaramella, Y.: Information retrieval and structured documents. In: Lectures on information retrieval, pp. 286–309 (2001)
5. Clarke, C.L.A.: Controlling overlap in content-oriented XML retrieval. In: ACM SIGIR 2005, pp. 314–321 (2005)
6. Fuhr, N., Lalmas, M., Malik, S., Kazai, G. (eds.): Advances in XML Information Retrieval and Evaluation (INEX 2005), Dagstuhl (2006)
7. Huibers, T.W.: An Axiomatic Theory for Information Retrieval. Universiteit Utrecht, PhD Thesis (1996)
8. Mass, Y., Mandelbrod, M.: Using the INEX environment as a test bed for various user models for XML retrieval. In: Fuhr, et al. (eds.) [6], pp. 187–195.
9. Sigurbjörnsson, B., Kamps, J.: The effect of structured queries and selective indexing on XML retrieval. In: Fuhr, et al. (eds.) [6], pp. 104–118.
10. van Rijsbergen, C.J.: Towards an information logic. In: ACM SIGIR 1989, pp. 77–86 (1989)
11. van Rijsbergen, C.J., Lalmas, M.: Information calculus for information retrieval. J. Am. Soc. Inf. Sci. 47(5), 385–398 (1996)
12. van Rijsbergen, C.J.v.: The Geometry of Information Retrieval. Cambridge University Press, Cambridge (2004)
13. Wong, K.-F., Song, D., Bruza, P., Cheng, C.-H.: Application of aboutness to functional benchmarking in information retrieval. ACM Trans. Inf. Syst. 19(4), 337–370 (2001)

An Effectiveness Measure for Ambiguous and Underspecified Queries

Charles L.A. Clarke, Maheedhar Kolla, and Olga Vechtomova

University of Waterloo, Canada

Abstract. Building upon simple models of user needs and behavior, we propose a new measure of novelty and diversity for information retrieval evaluation. We combine ideas from three recently proposed effectiveness measures in an attempt to achieve a balance between the complexity of genuine users needs and the simplicity required for feasible evaluation.

1 Introduction

A user types the query "windows" into a commercial Web search engine. She scans the result list. The first result is the Microsoft home page. It might contain the information she seeks, but she is not certain, and she moves on. The second result is the Vista home page. Probably not. The third result provides information about replacement windows and patio doors. No, not at all. The fourth page, a news article, provides most of the information she requires: Windows 7 will be released in November, and she can upgrade directly to it from XP. She navigates back to the initial result page and takes a quick glance at the fifth result, the Wikipedia page about MS Windows. She clicks on this link and never returns to the result list again.

When generating a ranked result, an information retrieval system should attempt to maximize the probability that a user will obtain the information she seeks. In our opening example, the IR system satisfied the user's requirements, but other users entering the same query will have other needs (and perhaps less patience). In generating a result list, the IR system must balance the requirements of the entire user population, reflecting the diversity of possible needs underlying the query and supplying novel information as users traverse the result list.

In this paper, we propose a new effectiveness measure, which evaluates the success of an IR system at achieving its goal of novelty and diversity. We measure success in terms of the expected number of relevant pages a user will encounter when scanning the results. In developing our measure, we build upon three recent proposals: i) the rank-biased precision measure proposed by Moffat and Zobel [1], ii) the α-nDCG measure proposed by Clarke et al. [2], and iii) the "intent aware" measures proposed by Agrawal et al. [3]. All three proposals are founded on simple models of user needs and behaviors. By combining them, we believe we can define a measure that achieves a balance between the complexity of genuine user needs and the simplicity required for feasible evaluation.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 188–199, 2009.

```
<topic number=0>
  <query> physical therapist </query>
  <description>
    The user requires information regarding the profession and the
    services it provides.
  </description>
  <subtopic number=1> What does a physical therapist do? </subtopic>
  <subtopic number=2> Where can I find a physical therapist? </subtopic>
  <subtopic number=3>
    How much does physical therapy cost per hour?
  </subtopic>
   ...
  <subtopic number=8>
    Information is required regarding physical therapist's assistants.
    What education do they require? How much do they make?
  </subtopic>
</topic>
```

Fig. 1. Example topic taken from the TREC 2009 Web Track guidelines. Some subtopics have been elided for conciseness.

In our proposal, we make a careful distinction between *ambiguity* and *underspecification*. An ambiguous query has multiple distinct *interpretations*. Interest in one interpretation suggests disinterest in the others. For example, the query "windows" may be related to the commercial software product or to the architectural feature, but probably not to both. An underspecified query has multiple *aspects*. A user might be interested in any of these aspects, independent of her interest in the others. For example, a user issuing the query "windows" might be interested in any (or all of) using, upgrading, updating, installing, configuring, or troubleshooting any of the current, past or future versions of the operating system. Of course, as illustrated by our example, a query may be both ambiguous and underspecified. Moreover, almost any query could be considered underspecified to some extent.

Novelty and diversity has received recent attention from a number of researchers. Most notably, Spärck Jones et al. [4] call for the creation of evaluation methodologies and test collections that incorporate diversity. Space limitations prevent us from including a full survey of the area, but both Clarke et al. [2] and Agrawal et al. [3] include summaries of prior work.

2 Novelty and Diversity

Clarke et al. [2] develop and evaluate an effectiveness measure (α-nDCG) that directly accommodates novelty and diversity. They primarily consider what we call underspecified queries (although they do not make our careful distinction between ambiguity and underspecification). They base their measure on the notion of a *nugget*, which they define very broadly as "any binary property of a document". Their nuggets essentially represent concrete instances of what we call aspects. We find it helpful to retain a distinction between aspects and nuggets. Aspects are conceptual; nuggets are operational. Nuggets can be assessed as part of an evaluation experiment. Aspects cannot.

Their definition of a nugget is general enough to encompass document properties such as answers to factual questions, transactional/navigational intent, and topical relevance. Nuggets related to our introductory example include the properties: i) "describes the procedure for upgrading from XP to Windows 7"; ii) "provides pricing and ordering information for replacement windows"; or iii) "is the Microsoft home page". We say a document "contains" a nugget if satisfies the associated binary property. Many documents contain the first two nuggets. Only one page can contain the third nugget.

The notion of a nugget (usually in a less general form) appears in many recent evaluation efforts, particularly in the areas of summarization and complex question answering [5,6]. Figure 1 shows an example topic taken from the guidelines of the ongoing TREC 2009 Web Track[1]. The topic includes a number of subtopics, each related to the broader topic, together with an associated query. As indicated in the track guidelines, the selection of subtopics is intended to comprise a representative, but not exhaustive, list of possible subtopics. These subtopics, which are engineered to be roughly balanced with respect to their relative popularity, were derived from co-clicks and other statistics extracted from the logs of a commercial Web search engine.

The subtopics are defined for judging purposes. Track participants execute the query and return results that attempt to cover as many of the (unseen) subtopics as possible. Documents are judged in terms of overall topical relevance, as well as with respect to the individual subtopics. Thus, in our view, we may associate a nugget with overall topical relevance, as well as with each subtopic. The track organizers plan to apply α-nDCG, intent aware nDCG, and other measures to evaluate effectiveness.

2.1 Diversity

Nuggets link users and documents. Consider a user u sampled from the population of all users entering a query q. Clarke et al. model the user's information need in terms of a set of nuggets

$$\mathcal{N} = \{n_1, n_2, ... n_N\}. \quad (1)$$

The probability that the user u is interested in a document containing the ith nugget is expressed as $\Pr[n_i \in u]$. The probability that the document d contains the ith nugget is expressed as $\Pr[n_i \in d]$.

Clarke et al. assume that a user's interest in one nugget is independent of her interest in other nuggets, implying that $\Pr[n_i \in u]$ can be estimated independently for each nugget. This assumption corresponds to our assumption of independence between aspects of an underspecified query. On the other hand, an assumption of independence is not appropriate when a query is ambiguous. If

[1] plg.uwaterloo.ca/~trecweb/; accessed April 17, 2009

a user is interested in buying windows for their house, we might guess that they are not interested in the Windows operating system, at least at that instant.

For a particular nugget n_i, the value of $\Pr[n_i \in u]$ depends on characteristics of the user population, which might be inferred from query logs and other sources of implicit feedback. Following the simplest approach, Clarke et al. assume

$$\Pr[n_i \in u] = \gamma, \tag{2}$$

for all nuggets n_i, where γ is a constant, $0 < \gamma \leq 1$. In light of our ability to balance relative popularity when choosing nuggets, as illustrated in Figure 1, this assumption appears reasonable.

Clarke et al. base $\Pr[n_i \in d]$ on explicit judgments. Since a nugget represents a binary property of a document, in theory d either contains or does not contain the nugget. However, the user may not recognize that the document satisfies her information need. Thus, $\Pr[n_i \in d]$ reflects a user's ability to judge whether or not the document contains the nugget, and her ability to extract the information if it does. Clarke et al. provide justification, and additional discussion, regarding this and other features of their model. Their paper should be consulted for a detailed explanation.

As part of an evaluation experiment, assessors judge whether or not document d contains nugget n_i. If document d is judged to contain nugget n_i then $\Pr[n_i \in d] = \alpha$, where α is a constant, $0 < \alpha \leq 1$. Otherwise, $\Pr[n_i \in d] = 0$. Now, if we let $J(d, i) = 1$ if d is judged to contain nugget n_i, and $J(d, i) = 0$ if it is not, we have

$$\Pr[n_i \in d] = \alpha J(d, i). \tag{3}$$

We consider document d to be relevant if the document contains any nugget that interests the user u. Thus, the probability of relevance $r(u, d)$ may be calculated as

$$\begin{aligned} r(u,d) &= 1 - \prod_{i=1}^{N} (1 - \Pr[n_i \in u] \cdot \Pr[n_i \in d]) \qquad (4) \\ &= \sum_{i=1}^{N} (\Pr[n_i \in u] \cdot \Pr[n_i \in d]) - O\left(\max_{1 \leq i \leq N} (\Pr[n_i \in u] \cdot \Pr[n_i \in d])^2\right) \\ &\approx \sum_{i=1}^{N} \Pr[n_i \in u] \cdot \Pr[n_i \in d]. \end{aligned}$$

Substituting Equations 2 and 3 into this last equation gives

$$r(u,d) \;=\; \gamma\alpha \sum_{i=1}^{N} J(d,i). \tag{5}$$

Thus, the probability of relevance is proportional to the number of nuggets a document contains.

2.2 Novelty

Nuggets reflect the diversity of information needs underlying a query. In doing so, they also allow us to measure novelty in a result list.

Equation 4 estimates a document's probability of relevance in isolation. Instead, consider the probability of relevance for a document d_k appearing in the context of a ranked list

$$\langle d_1, d_2, ..., d_{k-1}, d_k, ...\rangle. \tag{6}$$

Assume the user scans this list in order. If a document covers a particular aspect, Clarke et al. assume the user is less interested in seeing this information repeated in later documents. The probability that the user is interested in d_k because it contains nugget n_i now depends on the contents of the documents which precede it

$$\begin{aligned} \Pr[n_i \in u | d_1, d_2, ..., d_{k-1}] &= \Pr[n_i \in u] \prod_{j=1}^{k-1} \Pr[n_i \notin d_j] \\ &= \Pr[n_i \in u] \prod_{j=1}^{k-1} \left(1 - \Pr[n_i \in d_j]\right). \end{aligned} \tag{7}$$

Now define

$$C(k,i) = \begin{cases} \sum_{j=1}^{k-1} J(d_j, i) & \text{if } k > 1, \\ 0 & \text{if } k = 1. \end{cases} \tag{8}$$

$C(k,i)$ is the number of documents above rank k that have been judged to contain nugget n_i. Combining Equation 8 with Equation 3 gives

$$\prod_{j=1}^{k-1} (1 - \Pr[n_i \in d_j]) = (1-\alpha)^{C(k,i)}. \tag{9}$$

Substituting Equation 2 and Equation 9 into Equation 7 yields

$$\Pr[n_i \in u | d_1, d_2, ..., d_{k-1}] = \gamma (1-\alpha)^{C(k,i)}. \tag{10}$$

Finally, building on Equation 4, we estimate the probability of relevance for the kth document as

$$\sum_{i=1}^{N} \Pr[n_i \in u | d_1, d_2, ..d_{k-1}] \cdot \Pr[n_i \in d] = \alpha\gamma \sum_{i=1}^{N} J(d_k, i)(1-\alpha)^{C(k,i)}. \tag{11}$$

Thus, the probability of relevance is proportional to the number of nuggets a document contains, discounted according to the nuggets appearing at higher ranks. Clarke et al. drop the constant of proportionality, expressing the relevance of a retrieval result as a *gain vector*

$$\mathcal{G} = \langle g_1, g_2, ..., g_k, ...\rangle, \tag{12}$$

where

$$g_k = \sum_{i=1}^{N} J(d_k, i)(1 - \alpha)^{C(k,i)}. \tag{13}$$

They retrofit this gain vector into the Normalized Discounted Cumulative Gain (nDCG) measure of Järvelin and Kekäläinen [7] to produce their α-nDCG measure. Calculation of nDCG (and α-nDCG) requires the computation of an *ideal* gain vector

$$\mathcal{G}' = \langle g'_1, g'_2, ..., g'_k, ...\rangle. \tag{14}$$

This ideal gain vector corresponds to the document ordering that maximizes cumulative gain ($\sum_{j=1}^{k} g_j$) at every retrieval depth k. As we might expect, the ideal gain vector has the property that the gain values decrease monotonically ($g'_k \geq g'_{k+1}, \forall g'_k$).

3 Rank-Biased Precision

We extend the reasoning of the previous section to create a new effectiveness measure that rewards novelty. Rather than building on nDCG, we base our measure on the *rank-biased precision measure* (RBP) described by Moffat and Zobel [1].

3.1 User Model

The RBP user model assumes that the user, after issuing a query, begins reading the list of results from the top. After reading the first result, the user moves on to read the second result with constant probability β and stops reading altogether with probability $1 - \beta$. After reading the second result, assuming that she does, she moves on to read the third result with probability β and stops reading with probability $1 - \beta$. On so on. After reading result k, she will move on to the next result with probability β and stop reading with probability $1 - \beta$.

Adjusting the value of β allows us to adjust the model to reflect patient and impatient users. Higher values represent more patient users; lower values represent less patient users. At one extreme, if $\beta = 0$, the user never looks at anything other than the first document. At the other extreme, if $\beta = 1$, the user reads document after document, forever. It is important to note that the user's decision to move on is made independently of the relevance of the current document. Moffat and Zobel provide justification, and additional discussion, regarding this and other properties of their model. Their paper should be consulted for a detailed explanation.

The model implies that we may calculate the expected number of documents read by the user as

$$\sum_{k=1}^{\infty} \beta^{k-1} = \frac{1}{1 - \beta}. \tag{15}$$

3.2 Binary Relevance

Assume the relevance of the result list for a particular query is described by the vector

$$\mathcal{R} = \langle r_1, r_2, r_3, ...\rangle. \tag{16}$$

For now, we assume binary relevance judgments, with $r_k = 1$ indicating that the result at rank k is relevant, and $r_k = 0$ indicating that the result at rank k is non-relevant. Under the user model of Moffat and Zobel, we can calculate the expected number of relevant documents a user will encounter as

$$\sum_{k=1}^{\infty} r_k \beta^{k-1}. \tag{17}$$

This value forms the basis of rank-biased precision. The more relevant results encountered, the better the result list.

We may normalize this expected value to fall between zero and one by considering the "ideal" result list, which would consist of all the relevant documents in the corpus ranked before all non-relevant documents.

$$\mathcal{R}' = \langle 1, 1, 1, ..., 1, 0, 0, 0, ...\rangle. \tag{18}$$

If we assume there are R relevant documents in the corpus, then the kth value of $\mathcal{R}'$ equals 1 if $k \leq R$, and 0 otherwise. If a user is presented with this ideal result list, we can calculate the number of relevant results we expect her to encounter as

$$\sum_{k=1}^{R} \beta^{k-1} = \frac{1-\beta^R}{1-\beta}. \tag{19}$$

Since we can never do better than this ideal outcome, we can normalize Equation 17 by dividing by Equation 19 to give

$$\frac{1-\beta}{1-\beta^R} \sum_{k=1}^{\infty} r_k \beta^{k-1}. \tag{20}$$

Applying this equation requires us to determine a value for R. Unfortunately, determining this value can be difficult, especially over a large corpus. In theory, determining a value for R requires us to judge every document in the corpus. In practice, the pooling method makes a guess by assuming that documents outside the pool are non-relevant. As noted by Moffat and Zobel, the value of R plays a substantial role in the computation of several standard effectiveness measures, including average precision and bpref. They question the appropriateness of this role, since we might expect a user to be more interested in the relevant documents she actually encounters than in the (effectively unknowable) number of relevant documents that exist.

Fortunately, unless the total number of relevant documents is very small, R does not have a substantial impact on the value of Equation 20. Instead of our ideal vector, which depends on R, we can imagine an "ideal ideal" result vector, consisting of an infinite number of relevant documents, drawn from an infinite collection

$$\mathcal{R}'' = \langle 1, 1, 1, ... \rangle. \tag{21}$$

Adopting this ideal ideal result vector is equivalent to taking the limit as $R \to \infty$ of Equation 20 giving

$$\text{RBP} = (1 - \beta) \sum_{k=1}^{\infty} r_k \beta^{k-1}, \tag{22}$$

which is Moffat and Zobel's definition of rank-biased precision. In practice, the formula would be evaluated only to some predetermined maximum retrieval depth, but we see no pressing reason to explicitly introduce this depth into the formula.

In their presentation of RBP, Moffat and Zobel move straight from Equation 17 to Equation 22, skipping Equation 20 and avoiding any consideration of R. We take this extra step to highlight a connection with nDCG, which we further explore in Section 4. Moffat and Zobel advertise this independence from R as an important feature of RBP.

3.3 Probabilistic Relevance

We may extend RBP to probabilistic relevance values by simply replacing the binary relevance vector of Equation 16 with a probabilistic relevance vector.

$$\mathcal{R} = \langle r_1, r_2, r_3, ... \rangle. \tag{23}$$

where $0 \le r_k \le 1$ is the probability that a user will consider the kth document to be relevant to her specific combination of information needs. Under this extension, and assuming the RBP user model, the expected number of relevant results seen by the user is exactly Equation 17, unchanged:

$$\sum_{k=1}^{\infty} r_k \beta^{k-1}. \tag{24}$$

Normalization is a little more complex. Following the probability ranking principle, the ideal ranking of the documents in the collection is

$$\mathcal{R}' = \langle r'_1, r'_2, r'_3, ... \rangle, \tag{25}$$

where ${r_k}' \ge {r_{k+1}}'$, $\forall {r_k}'$. Normalizing Equation 24 using this ideal ranking gives

$$\frac{\sum_{k=1}^{\infty} r_k \beta^{k-1}}{\sum_{k=1}^{\infty} {r_k}' \beta^{k-1}}. \tag{26}$$

Of course, we can again replace this ideal result vector with an ideal ideal result vector (identical to Equation 21), reducing Equation 26 to Equation 22. Thus, extending RBP to accommodate probabilistic relevance values changes only the interpretation and usage of Equation 22, nothing more.

4 Novelty- and Rank-Biased Precision

In this section, we combine the ideas of Clarke et al. [2], as considered in Section 2, with the ideas of Moffat and Zobel [1], as considered in Section 3 to create a new measure, which we call *novelty- and rank-biased precision* (NRBP).

The elements of the gain vector in Equation 12, and the elements of the ideal gain vector in Equation 14, are proportional to the probability of relevance for the corresponding documents, with a constant of proportionality $\gamma\alpha$. Substituting into Equation 26 gives

$$\frac{\sum_{k=1}^{\infty} g_k \beta^{k-1}}{\sum_{k=1}^{\infty} {g_k}' \beta^{k-1}}, \tag{27}$$

where the constant of proportionality cancels.

This formula bears more than a passing resemblance to nDCG, incorporating such features as cumulative gain, discounts for retrieval depth, and normalization by an ideal gain vector. However, unlike nDCG, the computation of gains and discounts is motivated and derived directly from simple user models. Moreover, unlike nDCG, the measure is not reported "at a particular depth"; it is a true summary measure.

Unfortunately, computing the ideal gain vector poses its own problems. Given a set of judged documents, the computation is NP-Complete, although an acceptable approximation method is available [2,8]. More importantly, computation of the ideal gain vector theoretically requires judgments over the entire collection.

As we did in Sections 3.2 and 3.3, we may approximate this ideal vector with an ideal ideal vector. In this case, each element of the vector represents a document containing all of the N nuggets.

$$\mathcal{G}'' = \langle {g''}_1, {g''}_2, {g''}_3, ...\rangle = \langle N, (1-\alpha)N, (1-\alpha)^2 N, ...\rangle. \tag{28}$$

Now,

$$\sum_{k=1}^{\infty} {g_k}'' \beta^{k-1} = N \sum_{k=1}^{\infty} \left((1-\alpha)\beta\right)^{k-1} = \frac{N}{1-(1-\alpha)\beta}. \tag{29}$$

This last equation highlights an interesting commonality between the ideas of Clarke et al. and those of Moffat and Zobel. Both α and β, in some sense, reflect the user's declining interest as she reads down the list, either because she finds what she seeks or because she loses patience. Substituting Equation 29 for the denominator in Equation 27, and substituting Equation 13 into the numerator, produces our novelty- and rank-biased precision (NRBP) measure:

$$\text{NRBP} = \frac{1-(1-\alpha)\beta}{N} \sum_{k=1}^{\infty} \beta^{k-1} \sum_{i=1}^{N} J(d_k, i)(1-\alpha)^{C(k,i)}. \tag{30}$$

Note that normalization factor includes division by the number of nuggets, allowing us to average the measure across multiple topics, even when the topics have different numbers of nuggets.

5 Ambiguous and Under-Specified Queries

Agrawal et al. [3] develop and evaluate a number of effectiveness measures that directly accommodate novelty and diversity. They primarily consider what we call ambiguous queries (although they do not make our careful distinction between ambiguity and underspecification). They assume that both queries and documents belong to one or more *categories*, which are operational equivalents of what we call interpretations. As a final step, we extend the NRBP measure to accommodate ambiguity.

Categories link users and documents. A user entering a query is interested in documents belonging to only one category. For example, the query "windows" might be associated with the categories "computer software" and "building supplies". A user entering this query is interested only in one or the other, not both. These categories contrast with the nuggets of Clarke et al., who assume that a user's interest in one nugget is independent of her interest in other nuggets.

Agrawal et al. assume the existence of a known probability distribution that specifies the query category. For example, it may be that 90% of users entering the query "windows" are interested in computer software, while 10% are interested in building supplies. Using this distribution, Agrawal et al. describe a generic approach for adapting existing effectiveness measures to take query and document categories into account.

Assume a query may belong to one of M categories, with associated probabilities $p_1, p_2, ..., p_M$, where $\sum_{j=1}^{M} p_j = 1$. To compute an effectiveness measure according to their approach, a result list is judged M times, once with respect to each category. A separate effectiveness score S_j, $1 \leq j \leq M$ is determined for each category. Agrawal et al. then define what they call an *intent aware* version of the effectiveness measure as the weighted average of the individual effectiveness scores

$$\sum_{j=1}^{M} p_j S_j. \tag{31}$$

Applying this generic approach, they define intent aware versions of nDCG, average precision, and reciprocal rank.

We apply their generic approach to our NRBP measure. We start by assuming each of the M categories has a number of nuggets associated with it, where N_j is the number of nuggets associated with category j. Following from the reasoning of Section 4, the expected number of relevant documents seen by a user may then be estimated as proportional to

$$(1-(1-\alpha)\beta) \sum_{k=1}^{\infty} \beta^{k-1} \sum_{j=1}^{M} \frac{p_j}{N_j} \sum_{i=1}^{N_j} J(d_k, j, i)(1-\alpha)^{C(k,j,i)}. \tag{32}$$

$J(d_k, j, i) = 1$ if document d_k is judged to contain nugget i of category j; otherwise, $J(d_k, j, i) = 0$. $C(k, j, i)$ is the number of documents above rank k that have been judged to contain nugget i of category j.

```
<topic number=0>
  <query> windows </query>
  <category number=1 probability=0.90>
    <description>the Microsoft Windows operating system </description>
    <subtopic number=1>What's the URL for updating windows?</subtopic>
    <subtopic number=2>When will Windows 7 be available?</subtopic>
    <subtopic number=3>
      Can I upgrade directly from XP to Windows 7?
    </subtopic>
  </category>
  <category number=2 probability=0.10>
    <description>house windows </description>
    <subtopic number=1>Where can I purchase replacements?</subtopic>
    <subtopic number=2>What are available brands?</subtopic>
  </category>
</topic>
```

Fig. 2. Hypothetical evaluation topic illustrating categories and subtopics

Our estimation formula has grown somewhat complex. However, we believe it remains feasible to structure an evaluation exercise around this formula. Consider the (purely hypothetical) topic appearing in Figure 2. In this example, subtopics are grouped into categories, each with a corresponding probability. Like the example in Figure 1, the subtopics might be derived from search engine logs, and as suggested by Agrawal et al., so might the categories and probabilities. Judging would require only an overall topical relevance judgment for each category, along with a judgment for each subtopic in a relevant category. Given the constraint that a document may be relevant to only a single category, we expect judging effort to be similar to that required for the topic in Figure 1.

Before averaging Equation 32 across a set of topics, a final normalization step is required (a step overlooked by Agrawal et al.). Since no document may be relevant to more than one category, the value of the formula cannot equal one, even under ideal circumstances, unless only one category has a non-zero probability. To determine this final normalization factor, we again imagine an ideal result, which maximizes the value of Equation 32. While the value of this normalization factor depends on the category distribution, and does not have a simple closed-form solution, we may easily calculate it by simulating an ideal result.

To simulate an ideal result, we start at the first rank and work towards lower ranks. At each rank k, we imagine a document containing all the nuggets from a single category chosen according to the formula

$$\operatorname*{argmax}_{1 \leq j \leq M} \left(p_j (1-\alpha)^{D(k,j)} \right), \tag{33}$$

where $D(k,j)$ is the number of (simulated) documents from category j appearing above rank k. For example, suppose there are two categories, A and B, with associated probabilities 90% and 10%. If $\alpha = 0.5$ and $\beta = 0.8$, the ideal result is

$$\langle A, A, A, A, B, A, B, A, B, ...\rangle. \tag{34}$$

When evaluated by Equation 32 this ideal result would produce a value of approximately 0.929. The equation would then be normalized by the inverse of this value.

Let $I(\alpha, \beta, p_1, ..., p_M)$ be the ideal value calculated by the procedure above. The final version of our NRBP measure is then:

$$\text{NRBP} \; = \; \frac{1-(1-\alpha)\beta}{I(\alpha,\beta,p_1,...,p_M)} \sum_{k=1}^{\infty} \beta^{k-1} \sum_{j=1}^{M} \frac{p_j}{N_j} \sum_{i=1}^{N_j} J(d_k, j, i)(1-\alpha)^{C(k,j,i)}. \tag{35}$$

6 Concluding Discussion

Building upon simple models of user needs and behavior, we propose a new measure of novelty and diversity for information retrieval evaluation. The success of our proposal depends both on the degree to which it reflects genuine user requirements and on the feasibility of applying it in an evaluation experiment. We have conducted a number of experiments to demonstrate its validity and feasibility, including experiments on the TREC question answering test collection employed by Clarke et al. Regrettably, we have no room available in this paper to present the details of these experiments. The ongoing TREC Web track, from which Figure 1 is taken, will provide another opportunity to validate the measure.

References

1. Moffat, A., Zobel, J.: Rank-biased precision for measurement of retrieval effectiveness. ACM Transactions on Information Systems 27, 1–27 (2008)
2. Clarke, C.L., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkann, A., Büttcher, S., MacKinnon, I.: Novelty and diversity in information retrieval evaluation. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 659–666 (2008)
3. Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S.: Diversifying search results. In: Proceedings of the Second ACM International Conference on Web Search and Data Mining, pp. 5–14 (2009)
4. Spärck Jones, K., Robertson, S.E., Sanderson, M.: Ambiguous requests: Implications for retrieval tests. SIGIR Forum 41(2), 8–17 (2007)
5. Lin, J., Demner-Fushman, D.: Will pyramids built of nuggets topple over? In: Proceedings of the Human Language Technology Conference, 383–390 (2006)
6. Nenkova, A., Passonneau, R., McKeown, K.: The pyramid method: Incorporating human content selection variation in summarization evaluation. ACM Transactions on Speech and Language Processing 4(2) (2007)
7. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems 20(4), 422–446 (2002)
8. Chen, H., Karger, D.R.: Less is more: Probabilistic models for retrieving fewer relevant documents. In: 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 429–436 (2006)

An Analysis of NP-Completeness in Novelty and Diversity Ranking

Ben Carterette

Dept. of Computer and Info. Sciences, University of Delaware, Newark, DE, USA
carteret@cis.udel.edu

Abstract. A useful ability for search engines is to be able to rank objects with *novelty* and *diversity*: the top k documents retrieved should cover possible interpretations of a query with some distribution, or should contain a diverse set of subtopics related to the user's information need, or contain nuggets of information with little redundancy. Evaluation measures have been introduced to measure the effectiveness of systems at this task, but these measures have worst-case NP-complete computation time. We use simulation to investigate the implications of this for optimization and evaluation of retrieval systems.

1 Introduction

There has recently been interest in designing retrieval systems to rank documents with novelty and diversity: the retrieved documents should cover some set of subtopics or cover different possible interpretations of a query [1,2,3,4,5,6,7]. Various evaluation measures have been proposed for this task: Zhai et al. introduced variations of recall and precision that count the number of unique subtopics retrieved [6], and Clarke et al. introduced a "nugget"-based version of DCG that penalizes systems for retrieving redundant subtopics [3]. In theory these measures can be used for optimization as well. They are based on a Cranfield-like setting in which assessors have annotated documents not only on their relevance but also with respect to subtopics, interpretations, or nuggets. The system is rewarded for finding documents that contain subtopics or nuggets that have not previously been seen in higher-ranked documents.

These measures have something in common: computing them is an NP-complete problem [1,6,3]. Let $\mathcal{S}$ be a set of subtopics, interpretations, nuggets, or facets, and let $\mathcal{C}$ be a corpus of documents in which each document $\mathcal{D}$ contains zero or more elements of $\mathcal{S}$. Those that contain zero elements are nonrelevant. All three of the measures listed above are based on comparing the number of subtopics retrieved up to some rank j to the maximum number that could have been retrieved at the same rank. Finding this maximum is an instance of Set Cover, one of Karp's original 21 NP-complete problems [8].

This paper is presented in two parts. The first considers the worst-case implications of optimizing to and evaluating with NP-complete effectiveness measures. The second uses simulations to draw conclusions about the implications in the average case.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 200–211, 2009.

2 Worst-Case Analysis

Let us first define our evaluation measures using the notation above, then show how each is NP-complete. For simplicity we will refer to elements of $\mathcal{S}$ as *subtopics*, though they need not literally be subtopics.

2.1 Evaluation Measures

We consider three measures from the literature: S-recall and S-precision, and nugget NDCG. Before defining them, let us follow Zhai et al. in defining $\textsc{minRank}(\mathcal{S}, k)$ as the size of the smallest subset of documents in $\mathcal{C}$ that could contain ("cover") at least k subtopics in $\mathcal{S}$ [6].[1] (We will use unadorned $\textsc{minRank}$ for the case where $k = |\mathcal{S}|$.) This is clearly an instance of Minimum Set Cover and therefore NP-complete in general.

S-recall. S-recall is defined as the number of subtopics retrieved up to a given rank j divided by the total number of subtopics (size of $\mathcal{S}$) [6]:

$$\textit{S-recall} = \frac{|\cup_{i=1}^{j} \mathcal{D}_i|}{|\mathcal{S}|}.$$

Computing S-recall at an arbitrary j is polynomial time; we only need count the unique subtopics retrieved. But because $|\mathcal{S}|$ could vary greatly from topic to topic, it is useful to look at S-recall at rank $k = \textsc{minRank}(\mathcal{S}, |\mathcal{S}|)$. Analogously to R-precision, S-recall at $\textsc{minRank}$ has a minimum value of 0 and a maximum of 1 for every topic. It is, however, NP-complete.

S-precision. Zhai et al. defined S-precision as the minimum rank at which a given recall value could optimally be achieved divided by the first rank at which the same recall value actually has been achieved [6]:

$$\textit{S-precision} = \frac{\textsc{minRank}(\mathcal{S}, k)}{j^*}, \text{ where } j^* = \min\{j \text{ s.t. } |\cup_{i=1}^{j} \mathcal{D}_i| \geq k\}.$$

This is equivalent to $\textsc{minRank}(\mathcal{S}, k)$ divided by the first rank by which at least k unique subtopics have appeared.

Nugget nDCG. Standard DCG calculates a gain for each document based on its relevance and a logarithmic discount for the rank it appears at [9]. The nugget version for diversity evaluation defines the gain of a document in terms of how often pieces of relevant information within it appear in documents ranked above it [3]. The gain is incremented by 1 for each new piece of information, and α^m $(0 \leq \alpha \leq 1)$ for a piece of information that has already been seen m times. Since DCG is unbounded, it is standard to normalize it by the maximum possible value it could have (given a perfect ranking of documents); this is called nDCG. Since nugget DCG continues to reward systems even as they retrieve

[1] Note that while Zhai et al. defined this quantity in terms of a recall value, we define it in terms of the number of subtopics. The definitions are functionally equivalent.

redundant material (but less so with each additional redundancy), computing the normalizing factor is not a simple instance of Minimum Set Cover. It can be reduced from Vertex Cover, however, and is therefore NP-hard [3].

These are all good measures. Our concern is at their boundaries: there may be topics that we cannot properly evaluate or optimize systems for. These cases cannot be averaged out; they will be a source of systemic error in our evaluations. Our goal is to begin to estimate how frequent such cases may be and what the implications of their existence are.

2.2 Approximability

An approximation algorithm is an efficiently-computable algorithm that gives an approximate solution to a hard problem. Approximation algorithms are typically evaluated by an *approximation ratio* expressed as the rate of growth of the ratio of the approximate solution to the optimal solution.

Evaluation. There is a simple greedy algorithm for calculating $\textsc{minRank}(\mathcal{S}, k)$ and the normalizing factor in nugget nDCG: first take the document that contains the most subtopics, then the document that contains the most subtopics that have not already been taken, and so on until k subtopics have been covered. This greedy approach is in fact roughly the best approximation that can be achieved for Set Cover. Feige showed that set cover is inapproximable within $(1 - \epsilon) \ln |\mathcal{S}|$ for $\epsilon > 0$ unless NP has quasi-polynomial algorithms [10]. The greedy algorithm has approximation ratio H_s, where $s = \max_{S \in \mathcal{S}} |S|$ and $H_n = \sum_{i=1}^{n} 1/i$; the fact that $H_s \leq 1 + \ln s$ gives the result.

While the approximated minRank or normalizing factor can therefore be quite bad, the situation is somewhat better for the measures themselves. The measures exhibit *submodularity*, which means they can be approximated within a constant factor of $1 - 1/e$ [1]. Intuitively, even if we are overestimating the denominator by a large factor, the fact that there is a limited number of subtopics means that the marginal error in the approximate value of S-recall or S-precision decreases as that factor increases.

Optimization. The optimization problem is to rank documents such that S-recall, S-precision, or nDCG are maximized. The standard principle for optimization in IR is the *Probability Ranking Principle*, which says that ranking documents in decreasing order of probability of relevance gives the optimal expected precision and recall (and therefore R-precision and average precision and other such measures) [11]. The PRP assumes that documents are relevant independently of one another, so it is not suitable for optimization of novelty or diversity rankings [12].

Instead, the optimization analog to the greedy algorithm for approximating evaluation measures is a greedy algorithm for ranking documents: given k ranked documents, the $k+1$st should be the one that is most likely to satisfy the greatest number of previously-unsatisfied subtopics. However, unlike the PRP, which maximizes precision and recall at *every* rank, a greedy document-by-document ranking principle cannot necessarily provide maximum S-recall or S-precision or nDCG at every rank. This follows from the NP-completeness of the evaluation problem; if this were possible, the problem would be solvable with the greedy algorithm. The worst case for optimization, then, is that the system is optimized at rank $1 + \log |\mathcal{S}|$ but not at any higher rank.

2.3 Example

Suppose there are 14 subtopics and 5 relevant documents (that is, five documents that contain at least one subtopic).[2] Documents contain subtopics as follows:

$$\mathcal{D}_1 = \{S_1, S_2\} \qquad \mathcal{D}_4 = \{S_1, S_3, S_4, S_7, S_8, S_9, S_{10}\}$$
$$\mathcal{D}_2 = \{S_3, S_4, S_5, S_6\} \qquad \mathcal{D}_5 = \{S_2, S_5, S_6, S_{11}, S_{12}, S_{13}, S_{14}\}$$
$$\mathcal{D}_3 = \{S_7, S_8, S_9, S_{10}, S_{11}, S_{12}, S_{13}, S_{14}\}$$

A greedy algorithm will always take $\mathcal{D}_3$ followed by $\mathcal{D}_2$ followed by $\mathcal{D}_1$. The optimal algorithm's selections will depend on the quantity being computed. Consider each of our evaluation measures:

1. S-recall at minRank: using the greedy algorithm to compute minRank will result in S-recall being evaluating at rank 3, while the optimal is at rank 2. The approximation ratio of minRank is therefore 3/2.
2. S-precision depends on being able to calculate minRank($\mathcal{S}, k$), where k is the number of unique subtopics observed. For $k = 7$ and $k = 8$, the greedy and optimal algorithms agree that minRank($\mathcal{S}, 7$) = minRank($\mathcal{S}$, 8) = 1. They also agree for $k = 12$ (the first two documents selected by the greedy algorithm): minRank($\mathcal{S}, 12$) = 2. But for $k = 14$ (in the two documents selected by the optimal algorithm) there is disagreement. The greedy approach says minRank($\mathcal{S}, 14$) = 3, while the optimal says minRank($\mathcal{S}, 14$) = 2.
3. The normalizing factor for nDCG presents a problem in that the optimal set of documents over which it is computed can depend on the rank. At rank 1, the best possible DCG is achieved with $\mathcal{D}_3$ ($DCG = 8/\log_2(2)$). But at rank 2, the best possible DCG is achieved with $\mathcal{D}_4, \mathcal{D}_5$ ($DCG = 7/\log_2(2) + 7/\log_2(3)$). The optimal set at rank 1 is not a subset of the optimal set at rank 2, and therefore unachievable by any ranking algorithm.

Now let us consider how the two types of evaluation interact with greedy optimization versus optimizing for S-recall at minRank. Assuming a system with perfect knowledge of subtopics, a greedy system will take $\mathcal{D}_3, \mathcal{D}_2, \mathcal{D}_1$, then either of $\mathcal{D}_4, \mathcal{D}_5$ followed by the other. The optimal system will take $\mathcal{D}_4, \mathcal{D}_5$ followed by $\mathcal{D}_3, \mathcal{D}_2, \mathcal{D}_1$. This is illustrated in Figure 1, along with the minRanks calculated by a greedy approach and an optimal approach.

Table 1 shows the complete set of evaluations for the two systems: greedy system with greedy evaluation; greedy system with optimal evaluation; optimal system with greedy evaluation; and optimal system with optimal evaluation. Note

[2] This example is derived from Wikipedia's page on Set Cover (http://en.wikipedia.org/wiki/Set_cover_problem).

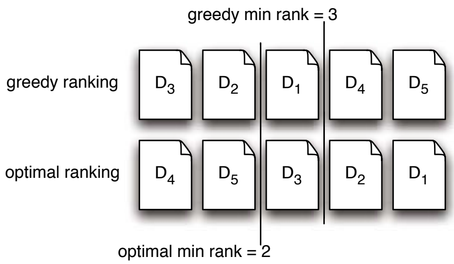


Fig. 1. A system that ranks documents greedily would place $\mathcal{D}_3$ above $\mathcal{D}_2$ above $\mathcal{D}_1$. A system that optimizes S-recall at $\textsc{minRank}(\mathcal{S})$ would place $\mathcal{D}_4, \mathcal{D}_5$ at the first two positions. Using a greedy algorithm to determine $\textsc{minRank}(\mathcal{S})$ places it at rank 3; the true value is at rank 2.

Table 1. Greedy and optimal evaluations for a system that ranks documents greedily and a system that optimizes for S-recall at the minimum rank

		greedy eval			optimal eval		
		rank 1	rank 2	rank 3	rank 1	rank 2	rank 3
greedy system	S-prec	1.000	1.000	1.000	1.000	1.000	1.000
	S-rec	0.571	0.857	1.000	0.571	0.857	1.000
	nDCG	1.000	1.000	1.000	1.000	0.922	0.859
optimal system	S-prec	1.000	1.333	1.333	1.000	1.000	1.000
	S-rec	0.500	1.000	1.000	0.500	1.000	1.000
	nDCG	0.875	1.085	1.164	0.875	1.000	1.000

that many of the values are greater than one for the optimal system evaluated greedily; this is because it is simply able to outperform the greedy algorithm.[3] Also note that even when evaluated optimally, the optimal system is outperformed at rank 1 by the greedy system; this is because, as mentioned above, the document that is optimal at rank one ($\mathcal{D}_3$) is not a subset of the documents that are optimal at rank two ($\mathcal{D}_4, \mathcal{D}_5$). Since the system is restricted to choosing a document at rank 1 that *is* a subset of the documents at ranks 1 and 2, it cannot optimize at *both* ranks and therefore must suffer at one of them.

The nDCG case is particularly interesting. We calculated nDCG with $\alpha = 1/2$, i.e. the second time a subtopic appears it contributes 1/2 to the document's gain, the third time it contributes 1/4, and so on. The system that optimizes S-recall therefore has incentive to go on to find the second-best set of documents and rank them second, thereby achieving an nDCG greater than 1 at both ranks 2 and 3 with the greedy evaluation. The greedy system evaluated optimally, on the other hand, sees a decrease in nDCG despite continuing to find novel subtopics; this is because it could have retrieved all 14 unique subtopics at rank 2, and 14 unique subtopics plus 8 redundant subtopics at rank 3.

[3] A simple "hack" for this case might be to redefine S-precision as $\min\{\textsc{minRank}(\mathcal{S}, k), j^*\}/j^*$, but this seems unfair to a system that actually is able to surpass the greedy algorithm.

The table shows that for optimization there is a firmly imposed tradeoff. When optimizing for S-recall at minRank, it is impossible to achieve perfect S-recall, S-precision, or nDCG at rank 1. When optimizing for S-precision or nDCG at each rank, it is impossible to achieve perfect S-recall at minRank. In standard retrieval problems founded on the PRP, there is an empirical tradeoff between precision and recall, but it is theoretically possible to optimize for both. For these measures there may be topics for which that is theoretically impossible; the developer is forced to choose.

This example can be generalized. If $|\mathcal{S}| = 2^{k+1} - 2$ and there are k relevant documents that are pairwise disjoint and $\mathcal{D}_i$ contains $2i$ subtopics, and there are two additional relevant documents that are disjoint and that each contain one half of each $\mathcal{D}_i$, the approximation ratio for minRank is $O(k/2)$. As k increases, the greedily-computed S-recall for a greedy system is 1, but the true S-recall is $(2^k + 2^{k-1})/(2^{k+1} - 2)$, which goes to $3/4$. Note that this is a constant approximation ratio for S-recall despite the logarithmic approximation ratio for minRank. This is due to the submodularity of S-recall [1].

3 Simulation and Analysis

While worst-case analysis shows that it is possible to construct cases in which the evaluation and optimization fail, the practical question is whether such cases occur in real data, and if so, how often and to what extent they affect evaluation and optimization. Having only a small sample of subtopic queries to analyze and no theory regarding the distribution of subtopics in documents, we cannot make definitive statements. But we can run simulations.

Due to space constraints, results in this section are reported exclusively for S-recall at minRank. S-recall is slightly simpler than S-precision and nDCG because it involves no parameters and is always between 0 and 1.

3.1 Real Data

There is little annotated data available for studying these problems. The largest set we are aware of is that constructed by Allan et al. a set of 60 topics with labeled "aspects" [13]. "Aspects" are defined as small pieces of relevant information; the system task was to retrieve as many unique aspects as possible in documents at the top of the ranking. For instance, the first query is "oil producing nations" and its relevant aspects are *Algeria, Angola, Azerbaijan, Bahrain, Brazil, Cameroon, Chad, China, ...*. Each document is labeled as to whether it is relevant to each of the topic's aspects. We obtained this data to use as a starting point; we will consider these aspects to be subtopics. We will consider each subtopic to be equally valuable to the user, so this problem is somewhat different from the diversity problems of Agrawal et al. and others that model a users' interest in particular subtopics.

Table 2 shows some example topics and their subtopics. Note that in some cases all subtopics are a particular entity type (dates, cities, etc), but in other cases they are not. Note also that there is much variance in the number of

Table 2. Examples of topics from the Allan et al. set

topic no.	query	# subtopics	# relevant docs
5	ohio highway shootings	33	52
	near I-270, near Columbus, a house, a freeway interchange, ...		
7	greenspan testimony congress	8	75
	Wed. Feb 11 2004, Thu. Feb 12 2004, Tue. Feb 24 2004, Apr 2004, ...		
18	haiti protest	7	48
	Port-au-Prince, Montreal, St. Marc, Raboteau, Gonaives, ...		
48	reduce dependence oil	17	12
	nuclear energy, shift to biodiesel, invest in hydrogen, ...		

subtopics and the number of relevant documents, and seemingly little correlation between the two. Many subtopics can occur in a single document, and a single subtopic can be duplicated in many documents.

Among these topics, there are two that are trivial: only one relevant document or only one subtopic. We have excluded these. Additionally, there are 27 (46.5%) that are quasi-trivial; in these, some subtopics only appear in one relevant document each, and taking those documents (and in some cases one additional document) covers the set trivially. There are four topics for which the greedy algorithm overestimates the true minRank. Therefore 4 out of 58 non-trivial topics (6.9%) and 13% of non-quasi-trivial topics can have performance overestimated by the greedy algorithm.

3.2 Simulated Topics

Starting from the real topics provided by Allan et al., we simulate new topics by sampling from a space defined by the marginal distributions of subtopics within documents. Specifically, each topic can be written as a matrix T with documents on the rows, subtopics on the columns, and $T_{ij} = 1$ if document i is relevant to subtopic j or $T_{ij} = 0$ otherwise. An example is shown in Table 3. We will simulate topics by sampling uniformly at random from the space of 0-1 matrices that have the same row sums and column sums as the initial topic matrix. This ensures that even if we cannot precisely model the distribution of subtopics in documents, we can at least model the numbers of subtopics contained in each document and the number of documents each subtopic appears in.

The sampling algorithm is based on a random walk procedure described by Zaman and Simberloff [14]. It is used in ecological studies for statistical testing of hypotheses about distributions of species in regions. It is based on the observation that within a larger matrix T, a 2×2 diagonal matrix $\left[\begin{smallmatrix}1&0\\0&1\end{smallmatrix}\right]$ can be changed to an anti-diagonal matrix $\left[\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right]$ (and vice versa) without altering the row or column sums. The algorithm works by sampling two rows and two columns uniformly at random, and if the 2×2 matrix formed from the cells at their intersections is diagonal or anti-diagonal, changing it to an anti-diagonal or diagonal matrix (respectively). Over many iterations this randomizes the distribution of subtopics in documents while keeping the marginal sums constant.

Table 3. Part of the document-subtopic matrix for topic 18 "haiti protest"

	Port-au-Prince	Montreal	St. Marc	Cap-Haïtien	Gonaives	Raboteau	Petionville	sum
$\mathcal{D}_1$	1	0	0	0	0	0	0	1
$\mathcal{D}_2$	1	0	0	0	0	0	0	1
$\mathcal{D}_3$	0	1	0	0	0	0	0	1
$\mathcal{D}_4$	0	0	0	1	0	0	0	1
$\mathcal{D}_5$	0	0	1	0	1	1	0	3
$\mathcal{D}_6$	1	0	1	0	1	0	0	3
...				...				...
sum	34	1	5	1	16	3	1	61

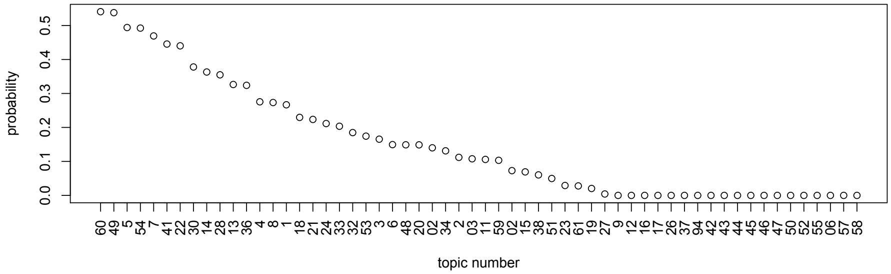


Fig. 2. Proportion of matrices sampled from the space defined by each of the baseline topics with MINRANK approximation ratio greater than 1

The algorithm requires a "burn-in" period to sufficiently randomize the original matrix. After that, a large enough number of sampling iterations ensures a uniform distribution over all possible matrices with the same row and column sums as the original. We used a burn-in period of 10,000 iterations, with 1,000 additional samples from the burned-in matrix to generate random topics. Thus for any given topic, we could generate a new random topic by iterating 1,000 times starting from the burned-in matrix for that topic.

Results. Results on simulated topics are based on evaluating a greedy system with perfect knowledge of subtopic containment. This is because the worst case for a system without perfect knowledge is arbitrarily bad: if such a system did not retrieve any relevant documents in the top $j =$ OPTIMAL-MINRANK, but it retrieved relevant documents at the following ranks up to $j =$ GREEDY-MINRANK, its S-recall approximation ratio goes to infinity. We consider simulated imperfect systems in the next section.

First we investigated the probability that the greedy algorithm for MINRANK would overestimate the minimum rank. Figure 2 shows the proportion of sampled matrices starting from each actual topic for which the true minimum rank (found by exhaustive search[4]) was less than the greedy minimum rank. Note for some

[4] Though this is a relatively small data set, exhaustive search still took a very long time in the most extreme cases, even when parallelized across 64 cores.

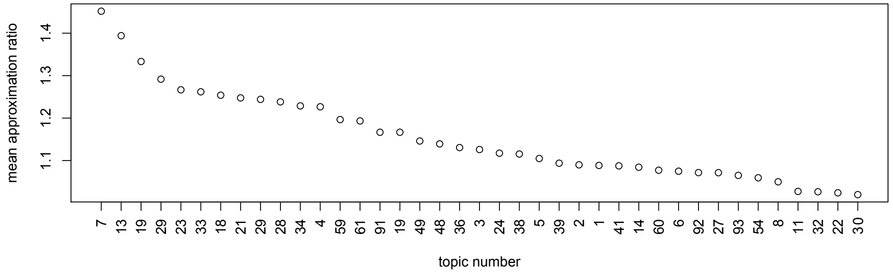


Fig. 3. Average minRank approximation ratio when greedy algorithm is suboptimal. Queries for which the greedy algorithm is always optimal not shown.

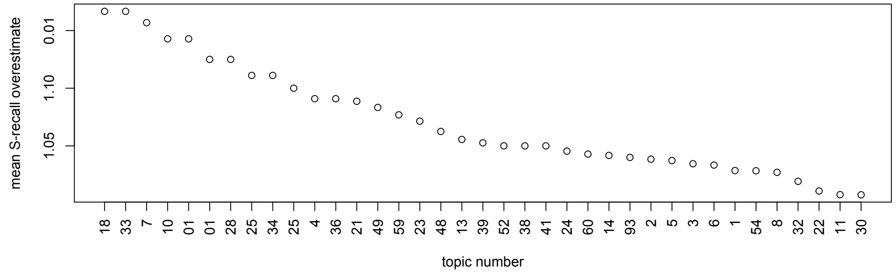


Fig. 4. Average factor by which S-recall is overestimated when greedy algorithm is suboptimal. Queries for which the greedy algorithm is always optimal not shown.

topics the probability is very high: for topic 60, over half the randomly sampled matrices were suboptimal.

There were 19 topics (roughly one third) for which the greedy and true minimum rank matched in every sample. Overall, the greedy algorithm overestimated minRank for about about 15% of sampled topics, which is a little higher than would be likely if the rate of 4 every 60 that was observed in the data is true.

Next we investigated the average minRank approximation ratio for the cases for which the greedy algorithm was suboptimal. Figure 3 shows the results for the 39 topics that were not always greedy-optimal. Topic 7 is the worst, with an average approximation ratio nearly 1.5 (minimum 1; maximum 1.667; median 1.333). Over all sampled topics, the mean approximation ratio is 1.16. The greedy is never more than 4 greater than the optimal, suggesting cases like our example above (worst case $\log |\mathcal{S}|$) are not occurring.

Finally we looked at the factor by which S-recall was overestimated when the rank was overestimated. Again, S-recall can only be overestimated by a constant $1 - 1/e$. Figure 4 shows that the average worst case is about 1.16 times the true value. The maximum factor by which any S-recall is overestimated is 1.33, which happens to be the reciprocal of the 3/4 approximation ratio derived in our example above.

3.3 Simulated Systems

As discussed above, the worst case for a system with perfect knowledge of subtopics is that S-recall is overestimated by a constant factor. The worst case for a system with no knowledge of subtopics (i.e. one that makes use of heuristics such as similarities between documents) is arbitrarily bad. Between these two extremes, we are interested in the cases of systems that use heuristics but that "look like" real systems might.

We simulated a "real" system that uses a greedy optimization approach as follows: starting with a document-subtopic matrix, we degraded it by changing each 1 indicating the presence of a subtopic i in a document j to a probability p_{ij} drawn from a Beta prior with parameters α_p, β_p. We changed each 0 indicating the absence of subtopic i in document j to a probability q_{ij} drawn from a Beta prior with parameters α_q, β_q. We then applied a greedy algorithm similar to Agrawal et al.'s IA-Select [1], which attempts to rank the documents that are most likely to satisfy previously-unsatisfied subtopics. The resulting ranked list is evaluated using S-recall.

The Beta distribution parameters $\alpha_p, \beta_p, \alpha_q, \beta_q$ offer some control over the expected quality of the simulated system. As $\alpha_p/(\alpha_p + \beta_p) \rightarrow 1$ and $\alpha_q/(\alpha_q + \beta_q) \rightarrow 0$, the system approaches perfection. As $\alpha_p/(\alpha_p + \beta_p) \rightarrow 0$ and $\alpha_q/(\alpha_q + \beta_q) \rightarrow 1$, the system approaches the worst possible. When $\alpha_p/(\alpha_p + \beta_p) = \alpha_q/(\alpha_q + \beta_q)$, the system is ranking documents randomly.

Results. To keep the parameter space manageable, we used $\alpha_p = \beta_q$ and $\alpha_q = \beta_p$, increasing α_p and α_q exponentially from 2^0 to 2^7. For large α_p and small α_q, the system is better; for small α_p and large α_q, the system is worse. At $\alpha_p = \alpha_q$ the performance is random.

We selected topics for which the greedy algorithms were suboptimal on either the burned-in matrix or the original matrix. We then degraded the matrix randomly and greedily re-ranked the documents according to the procedure above.[5] We then calculated S-recall both greedily and optimally.

Figure 5 compares the mean performance measured by the greedy evaluation to the S-recall approximation ratio for topics 5 and 7, starting from their burned-in matrices. Each point is the result of averaging over 100 trials with a particular α_p, α_q. Note that as simulated system performance degrades, we actually overestimate its performance more! This is quite disturbing, as it means that when the greedy evaluation is suboptimal, it will overestimate a bad system's performance more than a good system's performance. Bad systems will always appear better than they really are by a greater factor than good systems will.

The degree of overestimation is worse for topic 7 than for topic 5. This is because the optimal minimum rank for topic 7 is 3 (greedy is 4), while the optimal minimum rank for topic 5 is 16 (greedy is 18). With a deeper rank required for evaluation, the system has less opportunity to "catch up" after passing the optimal rank. However, topic 5 has five outlying points with very

[5] We did not do an optimal ranking, since there are too many documents to be able to do exhaustive search over all subsets.

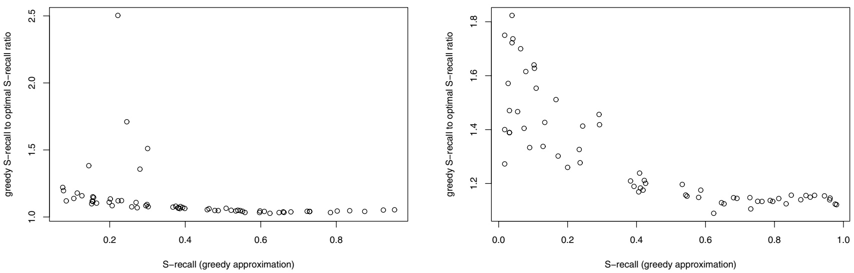


Fig. 5. Comparison of greedy S-recall to S-recall approximation ratio for topic 5 (left) and topic 7 (right) starting from burned-in matrices. Each point represents a different pair of prior parameters (α_p, α_q) and is averaged over 100 random trials.

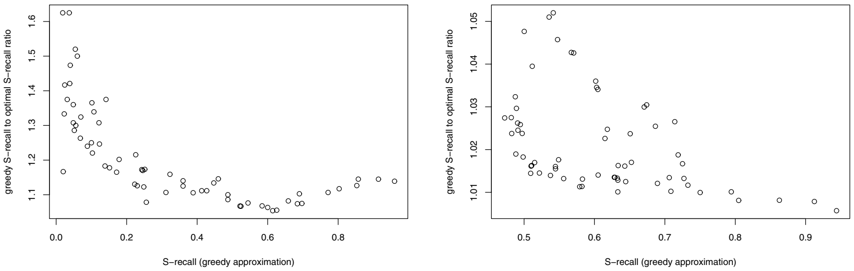


Fig. 6. Comparison of greedy S-recall to S-recall approximation ratio for topic 18 (left) and topic 30 (right) starting from original matrices. Each point represents a different pair of prior parameters (α_p, α_q) and is averaged over 100 random trials.

high approximation ratios. These are all points where α_q is substantially higher than α_p, meaning the system is *a priori* poor.

Figure 6 shows similar results starting from the original matrices for topics 18 and 30. Like topic 7, topic 18 has low optimal ranks (optimal 4 vs greedy 5). Like topic 5, topic 30 has high optimal ranks (optimal 53 vs greedy 52).

4 Conclusion

We have argued that NP-complete evaluation and optimization can be a serious problem for retrieval systems. Even if the approximation ratio is constant, we can significantly overestimate the performance of a system. These errors are not random errors that can be averaged out by sampling more topics; they are systemic problems with evaluation and optimization in this setting.

However, for many topics there is no problem. The greedy algorithm is optimal in 93% of the cases in "real" data, and in about 85% of cases in simulated data. The problem is those cases for which the greedy algorithm is not optimal, and in particular those cases in which a bad system is significantly overrated by the greedy algorithm, and those cases in which S-precision and nDCG cease to make sense as effectiveness measures. Future work should investigate characterizing the problematic topics so that results may be adjusted appropriately.

References

1. Agrawal, R., Gollapudi, S., Halverson, H., Ieong, S.: Diversifying search results. In: Proceedings of WSDM 2009, pp. 5–14 (2009)
2. Vee, E., Srivastava, U., Shanmugasundaram, J., Bhat, P., Amer-Yahia, S.: Efficient computation of diverse query results. In: Proceedings of ICDE 2008, pp. 228–236 (2008)
3. Clarke, C.L.A., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkan, A., Büttcher, S., MacKinnon, I.: Novelty and diversity in information retrieval evaluation. In: Proceedings of SIGIR 2008, pp. 659–666 (2008)
4. Radlinski, F., Kleinberg, R., Joachims, T.: Learning diverse rankings with multi-armed bandits. In: Proceedings of ICML 2008, pp. 784–791 (2008)
5. Chen, H., Karger, D.R.: Less is more: Probabilistic models for retrieving fewer relevant documents. In: Proceedings of SIGIR 2006, pp. 429–436 (2006)
6. Zhai, C., Cohen, W.W., Lafferty, J.D.: Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In: Proceedings of SIGIR 2003, pp. 10–17 (2003)
7. Carbonell, J.G., Goldstein, J.: The use of mmr, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of SIGIR 1998, pp. 335–336 (1998)
8. Garey, M.R., Johnson, D.S.: Computers and Intractibility: A Guide to the Theory of NP-completeness. W.H. Freeman, New York (1979)
9. Jarvelin, K., Kekalainen, J.: Cumulated gain-based evaluation of ir techniques. ACM Trans. Inf. Syst. 20(4), 422–446 (2002)
10. Feige, U.: A threshold of $\ln n$ for approximating set cover. Journal of the ACM 45(4), 634–652 (1998)
11. Robertson, S.E.: The probability ranking principle in information retrieval. Journal of Documentation 33, 294–304 (1977)
12. Goffman, W.: On relevance as a measure. Information Storage and Retrieval 2(3), 201–203 (1964)
13. Allan, J., Carterette, B., Lewis, J.: When will information retrieval be 'good enough?'. In: Proceedings of SIGIR 2005, pp. 433–440 (2005)
14. Zaman, A., Simberloff, D.: Random binary matrices in biogeographical ecology—instituting a good neighbor policy. Environmental and Ecological Statistics 9, 405–421 (2002)

From "Identical" to "Similar": Fusing Retrieved Lists Based on Inter-document Similarities

Anna Khudyak Kozorovitzky and Oren Kurland

Faculty of Industrial Engineering and Management, Technion, Haifa 32000, Israel
annak@techunix.technion.ac.il, kurland@ie.technion.ac.il

Abstract. We present a novel approach to *fusing* document lists that are retrieved in response to a query. Our approach is based on utilizing information induced from *inter-document similarities.* Specifically, the key insight guiding the derivation of our methods is that similar documents from different lists can provide *relevance-status* support to each other. We use a graph-based method to model relevance-status propagation between documents. The propagation is governed by inter-document-similarities and by retrieval scores of documents in the lists. Empirical evaluation shows the effectiveness of our methods in fusing TREC *runs.*

Keywords: fusion, inter-document-similarities, similarity-based fusion.

1 Introduction

The ad hoc retrieval task is to find the documents most pertaining to an information need underlying a given query. Naturally, there is a considerable amount of uncertainty in the retrieval process — e.g., accurately inferring the "actual" information need expressed by the query. Thus, researchers proposed to utilize different information sources and information types to address the retrieval task [1]. For example, utilizing multiple document representations, multiple query representations, and multiple search techniques have been proposed as a means to improving retrieval effectiveness [1].

Many of the approaches just mentioned depend on the ability to effectively *fuse* several retrieved lists so as to produce a single list of results. Fusion might be performed under a single retrieval system [2], or upon the results produced by different search systems (a.k.a. distributed/federated retrieval) [3,4]. Conceptually, fusion can be viewed as integrating "*experts' recommendations*" [1], where the expert is a retrieval model used to produce a ranked list of results — the expert's recommendation.

A principle underlying many fusion methods is that documents that are highly ranked in many of the lists, i.e., that are highly "recommended" by many of the "experts", should be ranked high in the final result list [3,5]. The effectiveness of approaches utilizing this principle often depends on the overlap between non-relevant documents in the lists being much smaller than that between relevant documents [5]. However, several studies have shown that this is often not the

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 212–223, 2009.

case, more specifically, that on many occasions there are (many) different relevant documents across the lists to be fused [6,7,8,9,10].

We propose a novel approach to fusion of retrieved lists that addresses, among others, the relevant-documents mismatch issue just mentioned. The key insight guiding the development of our methods is that *similar documents* from different lists can provide relevance-status support to each other, as they potentially discuss the same topics. Specifically, if relevant documents are assumed to be similar following the *cluster hypothesis* [11], then they can provide "support" to each other via inter-document similarities.

Our approach is based on using a graph-based method to model relevance-status propagation between documents in the lists to be fused. The propagation is governed by inter-document-similarities and by the retrieval scores of documents in the lists. Specifically, documents that are highly ranked in lists, and are similar to other documents that are highly ranked, are rewarded. If inter-document-similarities are not utilized — i.e., only retrieval scores are used — then some of our methods reduce to current state-of-the-art fusion approaches.

Empirical evaluation shows that our methods are effective in fusing high-quality TREC *runs*. Specifically, our most effective methods post performance that is superior to that of a state-of-the-art fusion method.

2 Fusion Framework

Notational conventions Let q and d denote a query and a document, respectively. We assume that documents are assigned with unique IDs; we write $d_1 \equiv d_2$ if d_1 and d_2 have the same ID, i.e., they are the same document. We assume that the document lists $L_1^{[q;k]}, \ldots, L_m^{[q;k]}$, or $L_1, \ldots, L_m$ in short, were retrieved in response to q by m retrievals performed over a given corpus, respectively; each list contains k documents. We write $d \in L_i$ to indicate that d is a member of L_i, and use $S_{L_i}(d)$ to denote the (positive) retrieval score of d in L_i; if $d \notin L_i$ then $S_{L_i}(d) \overset{def}{=} 0$. The *document instance* L_i^j is the document at rank j in list L_i. To simplify notation, we often use $S(L_i^j)$ to denote the retrieval score of L_i^j (i.e., $S(L_i^j) \overset{def}{=} S_{L_i}(L_i^j)$). The methods that we present consider the similarity $sim(d_1, d_2)$ between documents d_1 and d_2; we describe our similarity-induction method in Sect. 4.1.

2.1 Fusion Essentials

Our goal is to produce a single list of results from the retrieved lists $L_1, \ldots, L_m$. To that end, we opt to detect those documents that are "highly recommended" by the set $L_1, \ldots, L_m$, or in other words, that are "*prestigious*" with respect to this set. Given the virtue by which the lists were created, that is, in response to the query, we hypothesize that prestige implies relevance. The key challenge is then to formally define, and quantify, prestige.

Many current fusion approaches (implicitly) regard a document as prestigious if it is highly ranked in many of the lists. The CombSUM method [3], for example, quantifies this prestige notion by summing the document retrieval scores across the lists:

$$P_{CombSUM}(d) \overset{def}{=} \sum_{L_i : d \in L_i} S_{L_i}(d) \ .$$

To emphasize even more the importance of occurrence in many lists, the CombMNZ method [3,5], which is a state-of-the-art fusion approach, multiplies CombSUM's score by the number of lists a document is a member of:

$$P_{CombMNZ}(d) \overset{def}{=} \#\{L_i : d \in L_i\} \sum_{L_i : d \in L_i} S_{L_i}(d) \ .$$

An important source of information not utilized by current fusion methods is *inter-document relationships*. Specifically, documents that are similar to each other can provide support for prestige as they potentially discuss the same topics. Indeed, recent work on re-ranking a single retrieved list has shown that prestige, as induced from inter-document similarities, is connected with relevance [12]. In the multiple-lists setting that we address here, information induced from inter-document similarities across lists could be a rich source of helpful information as well. Case in point, a document that is a member of a single list, but which is similar to — and in the extreme case, a near-duplicate of — other documents that are highly ranked in many of the lists could be deemed prestigious. Furthermore, similarity-based prestige can be viewed as a generalization of the prestige notion taken by current fusion methods, if we consider documents to be similar if and only if they are the same document.

2.2 Similarity-Based Fusion

We use graphs to represent propagation of "prestige status" between documents; the propagation is based on inter-document similarities and/or retrieval scores. The nodes of a graph represent either documents, or document instances (appearances of documents) in the retrieved lists. In the latter case, the same document can be represented by several nodes, each corresponds to its appearance in a list, while in the former case, each node corresponds to a different document.

The following graph-construction method and prestige induction technique are inspired by work on inducing prestige in a single retrieved list [12]. Formally, given a set of documents (document instances) V, we construct a weighted (directed) complete graph $G \overset{def}{=} (V, V \times V, wt)$ with the edge-weight function wt:[1]

$$wt(v_1 \rightarrow v_2) \overset{def}{=} \begin{cases} sim(v_1, v_2) & \text{if } v_2 \in Nbhd(v_1; \alpha) \ , \\ 0 & \text{otherwise} \ ; \end{cases}$$

[1] Refer to [12,13] for discussion of the importance of directionality in graphs modeling inter-document-similarities.

$v_1, v_2 \in V$ and $Nbhd(v;\alpha)$ is the α elements v' in $V - \{v'' : v'' \equiv v\}$ that yield the highest $sim(v, v')$ — i.e., v's nearest neighbors in V. (α is a free parameter.)[2] Similar nearest-neighbor-based graph construction methods were shown to be effective for re-ranking a single list [14,12].

As in work on inducing, for example, (i) journal prestige in bibliometrics [15], (ii) Web-page prestige in Web retrieval [16], and (iii) plain-text prestige for re-ranking a single list [12], we can say that a node v in G is prestigious to the extent it receives prestige-status support from other prestigious nodes. We can quantify this prestige notion using $P(v;G) \stackrel{def}{=} \sum_{v' \in V} wt(v' \to v) P(v';G)$. However, this recursive equation does not necessarily have a solution.

To address this issue, we define a smoothed version of the edge-weight function, which echoes PageRank's [16] approach:

$$wt^{[\lambda]}(v_1 \to v_2) \stackrel{def}{=} \lambda \cdot \frac{\widehat{sim}(v_2, q)}{\sum_{v' \in V} \widehat{sim}(v', q)} + (1 - \lambda) \cdot \frac{wt(v_1 \to v_2)}{\sum_{v' \in V} wt(v_1 \to v')} \quad ; \qquad (1)$$

λ is a free parameter, and $\widehat{sim}(v, q)$ is v's estimated query-similarity. (Below we present various query-similarity measures.) The resultant graph is $G^{[\lambda]} \stackrel{def}{=} (V, V \times V, wt^{[\lambda]})$.

Note that each node in $G^{[\lambda]}$ receives prestige-status support to an extent partially controlled by the similarity of the document it represents to the query. Nodes that are among the nearest-neighbors of other nodes get an additional support. Moreover, $wt^{[\lambda]}$ can be thought of as a probability transition function, because the sum of weights on edges going out from a node is 1; furthermore, every node has outgoing edges to *all* nodes in the graph (self-loops included). Hence, $G^{[\lambda]}$ represents an ergodic Markov chain for which a unique stationary distribution exists [17]. This distribution, which can be found using, for example, the Power method [17], is the unique solution to the following prestige-induction equation under the constraint $\sum_{v' \in V} P(v'; G^{[\lambda]}) = 1$:

$$P(v; G^{[\lambda]}) \stackrel{def}{=} \sum_{v' \in V} wt^{[\lambda]}(v' \to v) P(v'; G^{[\lambda]}) \ . \qquad (2)$$

Algorithms. To derive specific fusion methods, we need to specify the graph $G^{[\lambda]}$ upon which prestige is induced in Eq. 2. More specifically, given the lists $L_1, \ldots, L_m$, we have to define a set of nodes V that represents documents (or document instances); and, we have to devise a query-similarity estimate ($\widehat{sim}(v, q)$) to be used by the edge-weight function $wt^{[\lambda]}$ from Eq. 1. The alternatives that we consider, which represent some of the ways to utilize our graph-based approach, and the resultant fusion methods are presented in Table 1. It is important to note that each fusion method produces a ranking of documents wherein a document cannot have more than one instance.

[2] Note that $Nbhd(v;\alpha)$ contains only nodes that represent documents *different* than that represented by v.

Table 1. Similarity-based fusion algorithms; $Score(d)$ is d's final retrieval score. Note that if document d appears in 3 document lists, for example, then it will be represented in V by (i) a single node under the "Set" representation, (ii) three nodes under the "Bag" representation, and (iii) nine nodes under the "BagDup" representation.

Algorithm	V	$\widehat{sim}(v,q)$	$Score(d)$
SetUni	$\{d : d \in \bigcup_i L_i\}$	1	$P(d; G^{[\lambda]})$
SetSum	$\{d : d \in \bigcup_i L_i\}$	$P_{CombSUM}(v)$	$P(d; G^{[\lambda]})$
SetMNZ	$\{d : d \in \bigcup_i L_i\}$	$P_{CombMNZ}(v)$	$P(d; G^{[\lambda]})$
BagUni	$\{L_i^j\}_{i,j}$	1	$\sum_{v \in V : v \equiv d} P(v; G^{[\lambda]})$
BagSum	$\{L_i^j\}_{i,j}$	$S(v)$	$\sum_{v \in V : v \equiv d} P(v; G^{[\lambda]})$
BagDupUni	$\{Dup(L_i^j)\}_{i,j}$	1	$\sum_{v \in V : v \equiv d} P(v; G^{[\lambda]})$
BagDupMNZ	$\{Dup(L_i^j)\}_{i,j}$	$S(v)$	$\sum_{v \in V : v \equiv d} P(v; G^{[\lambda]})$

The first group of methods does not consider occurrences of a document in multiple lists when utilizing inter-document similarities. Specifically, V, the set of nodes, is defined to be the set-union of the retrieved lists. Thus, each document is represented in the graph by a single node. The prestige value of this node serves as the final retrieval score of the document. The **SetUni** method, for example, ignores the retrieval scores of documents by using a uniform query-similarity estimate; hence, only inter-document similarity information is utilized. The **SetSum** and **SetMNZ** methods, on the other hand, integrate also retrieval-scores by using the CombSUM and CombMNZ prestige scores for query-similarity estimates, respectively.

The SetSum and SetMNZ algorithms are, in fact, generalized forms of CombSUM and CombMNZ, respectively. If we use the edge-weight function $wt^{[1]}$ (i.e., set $\lambda = 1$ in Eq. 1), that is, do not exploit inter-document-similarity information, then SetSum and SetMNZ amount to CombSUM and CombMNZ, respectively. (Proof omitted due to space considerations.) More generally, SetSum and SetMNZ control the reliance on retrieval scores versus inter-document similarities using the parameter λ.

In contrast to the first group of methods, the second considers occurrences of a document in multiple lists in utilizing inter-document similarity information. Specifically, each node in the graph represents an instance of a document in a list. Hence, the set of nodes in the graph (V) could be viewed as the bag-union of the retrieved lists. The final retrieval score of a document is set to the sum of prestige scores of the nodes that represent it — i.e., that correspond to its instances in the lists. It is also important to note that while the neighborhood set $Nbhd(v; \alpha)$ of node v cannot contain nodes representing the same document represented by v, it can contain multiple instances of a different document. Thus, documents with many instances receive more inter-document-similarity-based prestige-status support than documents with fewer instances.

The first representative of the bag-based algorithms, **BagUni**, ignores retrieval scores and considers only inter-document-similarities. Hence, BagUni differs from SetUni only by the virtue of rewarding documents with multiple instances. In addition to exploiting inter-document similarities, the

BagSum method also uses the retrieval score of a document instance as the query-similarity estimate of the corresponding node. We note that CombSUM is a specific case of BagSum with $\lambda = 1$, as was the case for SetSum. (Proof omitted due to space considerations.) Furthermore, BagSum resembles SetSum in that it uses λ for controlling the balance between using retrieval scores and utilizing inter-document similarities. However, documents with many instances get more prestige-status support in BagSum than in SetSum due to the bag-union representation of the lists.

Naturally, then, we opt to create a bag-based generalized version of the CombMNZ algorithm. To that end, for *each* document instance L_i^j that corresponds to document d, we define a new list $Dup(L_i^j)$. This list contains n copies of d, each assigned to an arbitrary different rank between 1 and n with $S(L_i^j)$ as a retrieval score; $n \overset{def}{=} \#\{L_i : d \in L_i\}$ — the number of original lists that d belongs to. The set of nodes V is composed of all document instances in the *newly* defined lists. The **BagDupUni** algorithm, then, uses a uniform query-similarity estimate. Hence, as SetUni and BagUni it utilizes only inter-document similarities; but, in doing so, BagDupUni rewards to a larger extent documents with multiple instances due to the bag representation and the duplicated instances. The **BagDupMNZ** algorithm integrates also retrieval-scores information by using the retrieval score of a document instance in a new list as the query-similarity estimate of the corresponding node. For $wt^{[1]}$ (i.e., $\lambda = 1$), BagDupMNZ amounts to CombMNZ, as was the case for SetMNZ. (Proof omitted due to space considerations.) Yet, BagDupMNZ rewards to a larger extent documents with multiple instances than SetMNZ does due to the bag representation of the lists and the duplicated document instances.

3 Related Work

Fusion methods usually use the ranks of documents in the lists, or their relevance scores, but not the documents' content (e.g., [3,5,1,18,19]), as opposed to our methods. By construction, some of our methods generalize such fusion methods, namely, CombSUM and CombMNZ [3]. We demonstrate the relative merits of our methods with respect to these fusion methods in Sect. 4.2. Also, we note that our methods can potentially utilize document *snippets* (i.e., summaries) for computing inter-document similarities, rather than the entire document content, if the content is not (quickly) accessible. Indeed, snippets were used for inducing inter-document similarities so as to cluster results of Web search engines [20]. Snippets (and other document features) were also utilized in some fusion models [21,22,23], but inter-document(snippet) similarities were not exploited.

There is a large body of work on re-ranking an initially retrieved list using graph-based methods that model inter-document similarities within the list (e.g., [24,14,12,25,26]). As mentioned in Sect. 2, our fusion methods could conceptually be viewed as a generalization of some of these approaches [24,14,12]; specifically, of methods that utilize both retrieval scores and inter-document-similarities for modeling relevance-status propagation within the list [24,14]. A similar relevance-status propagation method was also employed in work on sentence retrieval for question answering [27].

Methods utilizing inter-text similarities — some using a variant of PageRank as we do here — were also used, for example, for cross-lingual retrieval [28], prediction of retrieval effectiveness [29], and text summarization [30,31].

4 Evaluation

In what follows we explore the effectiveness (or lack thereof) of our similarity-based fusion methods.

4.1 Experimental Setup

To measure inter-document similarities, we use a previously-proposed language-model-based estimate [12]. Specifically, let $p_d^{[\mu]}(\cdot)$ denote the unigram, Dirichlet-smoothed, language model induced from document d, where μ is the smoothing parameter [32]. (We set $\mu = 1000$ following previous recommendations [32].) We define for documents d_1 and d_2:

$$sim(d_1, d_2) \overset{def}{=} \exp\left(-D\left(p_{d_1}^{[0]}(\cdot) \,\middle|\middle|\, p_{d_2}^{[\mu]}(\cdot)\right)\right) \quad ;$$

D is the KL divergence. This similarity measure was shown to be effective in previous work on re-ranking search results using graph-based methods [12,26].

For experiments we use TREC data sets, which were also used in some previous work on fusion (e.g., [18,19]); specifically, the ad hoc track of trec3, the web tracks of trec9 and trec10, and the robust track of trec12. We apply tokenization, Porter stemming, and stopword removal (using the INQUERY list) to the documents using the Lemur toolkit (www.lemurproject.org), which is also used for computing $sim(d_1, d_2)$.

Graph-based methods that utilize inter-document similarities for re-ranking search results are known to be most effective when employed over relatively short lists [14,12,26]. The methods are especially effective in improving precision at the very top ranks [12,26]. Hence, we take the following design decisions with respect to the number of lists to be fused (relatively small), the number of documents in each list (relatively small), and the evaluation measures that we focus on (measures of precision at top ranks).

We use our methods to fuse three lists, each of which corresponds to the top-k documents in a submitted run within a track. The three runs are the most effective among *all* submitted runs with respect to MAP@k (mean average non-interpolated precision at cutoff k, henceforth denoted MAP). The runs are denoted, by descending order of MAP performance, **run1**, **run2**, and **run3**, respectively. Thus, the initial ranking of the lists to be fused is of high quality. Experiments showed (actual numbers are omitted due to space considerations) that $k = 20$, which is used here and after, yields very good performance with respect to $k \in \{5, 10, 30, 40, 50\}$. This finding supports the observation from above with respect to the lengths of the lists to be fused.

It is important to note that fusing the three most effective runs does not constitute an attempt to devise a new fusion-based retrieval approach, since in "real life" no relevance judgments are available; rather, the idea is to study the potential effectiveness of our models in fusing high quality search results.

For inter-list compatibility of retrieval scores, we normalize the score of a document in a list with respect to the sum of all scores in the list. If a list is of negative retrieval scores, which is usually due to using logs, we use the exponent of a score for normalization[3].

We use the precision of the top 5 and 10 documents (p@5, p@10), and MAP(@k) for performance evaluation measures. We set the values of the free parameters of our methods to optimize p@5, following the previous findings described above with regard to precision-at-top-ranks effectiveness[4]. Specifically, the value of the ancestry parameter α is chosen from $\{5, 10, 20, 30, 40, 50\}$. (A relatively small value of α is often optimal.) The value of λ, which controls the reliance on retrieval scores versus inter-document-similarities, is chosen from $\{0.1, 0.2, \ldots, 1\}$; we study the effect of varying λ in Sect. 4.2. To determine statistically-significant performance differences, we use the two-tailed Wilcoxon test at the 95% confidence level.

For reference comparisons to our methods we use **optimized baselines** ("opt. base." in short): for each track and evaluation metric m, we report the best m-performance obtained by *any* submitted-run in this track. (Note that the MAP performance of the optimized baseline is that of run1 by the virtue of the way run1 was selected.) In addition, we compare our methods' performance with that of the CombSUM and CombMNZ fusion techniques; recall that these are special cases of some of our methods.

Efficiency Considerations. The number of documents (document instances) in the graphs we construct is at most a few hundreds[5]. Hence, if there is quick access to the documents' content, or alternatively, to document snippets — following the discussion in Sect. 3 — then computing inter-document similarities based on this information does not incur a significant computational overhead. Similar efficiency considerations were made in work on *clustering* the results retrieved by Web search engines [20]. In addition, we note that computing prestige over such small graphs takes only a few iterations of the Power method [17].

4.2 Experimental Results

Table 2 presents the performance numbers of the different methods. Our first observation is that integrating inter-document-similarities with retrieval scores

[3] Normalizing retrieval scores with respect to the maximum and minimum scores in a list yields almost exactly the same performance numbers as those we report here.

[4] If two parameter settings yield the same p@5, we choose the one *minimizing* p@10 so as to provide conservative estimates of performance; if there are ties for both p@5 and p@10, we choose the setting that minimizes MAP.

[5] Note that each of the three fused lists contains 20 documents, and each document instance is duplicated, if at all, at most three times.

from the lists results in performance that transcends that of using each alone. Indeed, the methods with the suffix "Uni" that use a uniform query-similarity estimate, i.e., that disregard retrieval scores in the lists, post performance that is almost always worse than that of their counterparts that do utilize retrieval scores for inducing query similarity. (Compare SetUni with SetSum and SetMNZ; BagUni with BagSum; and, BagDupUni with BagDupMNZ.) Furthermore, recall that the CombSUM and CombMNZ methods that utilize only retrieval scores are special cases of our "XSum" and "XMNZ" methods, respectively, if no inter-document-similarities are used. We can see that each of the "XSum" and "XMNZ" methods outperforms its special case (CombSUM and CombMNZ, respectively) in most relevant comparisons (track × evaluation metric), with several of the differences being statistically significant.

Moreover, the performance of the "XSum" and "XMNZ" methods that integrate retrieval scores with inter-document-similarities is almost always better — and in many cases to a statistically significant degree — than that of run2 and run3; the performance also transcends that of run1 and the optimized baselines,

Table 2. Performance numbers. The best result in a column is boldfaced. Statistically significant differences with the optimized baselines, run1, run2, and run3, are marked with 'o', 'a', 'b', and 'c', respectively. Statistically significant difference between our "XSUM" and "XMNZ" models and their "special cases", i.e., CombSUM and CombMNZ, respectively, are marked with 'm'.

	trec3			trec9		
	p@5	p@10	MAP	p@5	p@10	MAP
opt. base.	76.0	72.2	10.4	60.0	**53.1**	**28.2**
run1	74.4	72.2	10.4	60.0	**53.1**	**28.2**
run2	72.8	67.6	9.6	45.8^{o}	38.8^{o}	18.4^{o}
run3	76.0	71.2	9.5	38.3^{o}	34.6^{o}	16.8^{o}
CombSUM	80.8_{ab}	74.6_{b}	10.9_{bc}	52.9_{bc}	48.5_{bc}	24.9_{bc}
CombMNZ	80.8_{ab}	74.6_{b}	10.9_{bc}	55.0_{bc}	48.8_{bc}	25.5_{bc}
SetUni	79.2	75.0	10.4	42.5^{o}	39.2^{o}	16.1^{o}
SetSum	82.8^{o}_{abc}	78.0^{om}_{abc}	$\mathbf{11.5}^{om}_{abc}$	59.2^{m}_{bc}	49.2_{bc}	26.5^{m}_{bc}
SetMNZ	82.0_{ab}	77.2^{o}_{abc}	11.3^{o}_{abc}	$\mathbf{61.3}^{m}_{bc}$	49.2_{bc}	28.0^{m}_{bc}
BagUni	82.4_{ab}	78.8^{om}_{abc}	11.1_{bc}	59.2_{bc}	47.9_{bc}	24.1_{bc}
BagSum	$\mathbf{83.2}^{o}_{abc}$	78.8^{om}_{abc}	11.2^{o}_{abc}	59.6^{m}_{bc}	48.1_{bc}	24.6_{bc}
BagDupUni	82.0_{ab}	78.6^{o}_{abc}	11.3^{o}_{abc}	57.5_{bc}	48.1_{bc}	24.9_{bc}
BagDupMNZ	$\mathbf{83.2}_{ab}$	$\mathbf{79.0}^{om}_{abc}$	$\mathbf{11.5}^{om}_{abc}$	60.4^{m}_{bc}	47.9_{bc}	25.4_{bc}

	trec10			trec12		
	p@5	p@10	MAP	p@5	p@10	MAP
opt. base.	63.2	58.8	30.7	54.5	48.6	28.8
run1	63.2	58.8	30.7	51.1	44.8	28.8
run2	54.4	50.2	27.7^{o}	52.5	48.6	28.4
run3	55.6	46.8^{o}	21.6^{o}	51.5	45.2^{o}	28.1
CombSUM	71.2^{o}_{abc}	61.0_{bc}	$\mathbf{37.2}_{bc}$	53.7	$\mathbf{49.2}_{ac}$	$\mathbf{30.3}^{o}_{a}$
CombMNZ	71.2^{o}_{abc}	61.0_{bc}	$\mathbf{37.2}_{bc}$	53.9	$\mathbf{49.2}_{ac}$	$\mathbf{30.3}^{o}_{a}$
SetUni	56.8	48.2^{o}	24.4^{o}	47.3^{o}	41.5^{o}	25.8
SetSum	71.2^{o}_{abc}	61.0_{bc}	$\mathbf{37.2}_{bc}$	55.4_{a}	48.5_{ac}	30.1^{o}_{a}
SetMNZ	71.2^{o}_{abc}	61.0_{bc}	$\mathbf{37.2}_{bc}$	55.6_{ac}	48.5_{ac}	$\mathbf{30.3}^{o}_{a}$
BagUni	70.8_{bc}	$\mathbf{61.2}_{bc}$	35.6_{bc}	53.1	46.5	28.2
BagSum	71.2^{o}_{abc}	61.0_{bc}	$\mathbf{37.2}_{bc}$	55.4_{ac}	$\mathbf{49.2}_{ac}$	29.8^{o}_{a}
BagDupUni	$\mathbf{72.0}^{o}_{abc}$	60.4_{bc}	35.8_{bc}	52.9	47.8	28.4
BagDupMNZ	$\mathbf{72.0}^{o}_{abc}$	61.0_{bc}	36.7_{bc}	$\mathbf{56.6}^{m}_{abc}$	49.0_{ac}	30.1^{o}_{a}

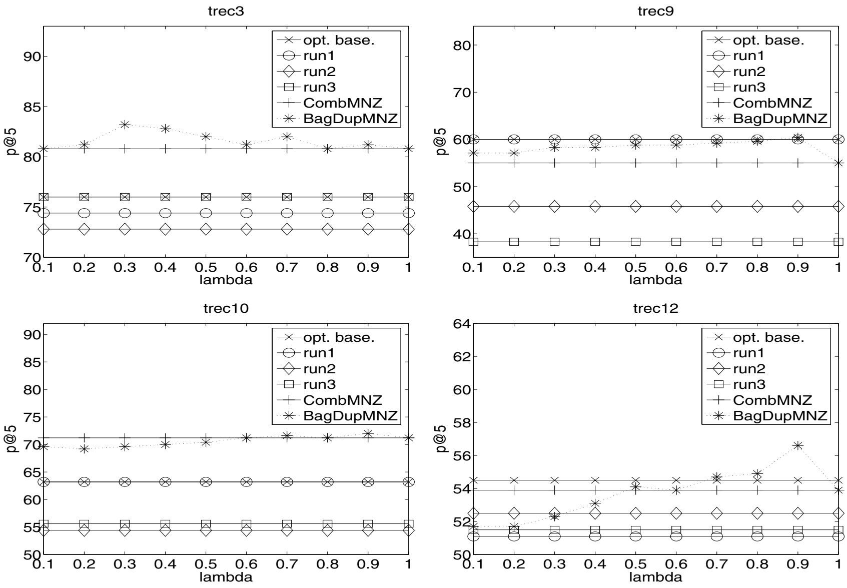


Fig. 1. Effect of varying λ (refer to Eq. 1 in Sect. 2) on the p@5 performance of BagDupMNZ; $\lambda = 1$ amounts to the CombMNZ algorithm. The performance of the optimized baseline, run1, run2, run3, and CombMNZ is depicted with horizontal lines for reference. Note: figures are not to the same scale.

except for trec9. (Note that for trec9 the performance of run1 is by far better than that of run2 and run3.) Thus, these findings attest to the merits of integrating retrieval scores and inter-document similarities for fusion — the underlying idea of our approach.

We can also see in Table 2 that the bag representation of the lists yields better performance, in general, than that of the set representation (e.g., compare BagUni with SetUni, and BagSum with SetSum). Hence, the fact that documents with occurrences in many of the fused lists can draw more prestige-status support via inter-document-similarities than documents with fewer occurrences (refer back to Sect. 2.2) has positive impact on performance.

Thus, it is not a surprise that the BagSum and BagDupMNZ methods that use a bag-representation of the lists, and that integrate retrieval scores with inter-document-similarities, are among the most effective similarity-based fusion algorithms that we consider. Specifically, BagDupMNZ posts the best p@5-performance (the metric for which performance was optimized) in Table 2 for three out of the four tracks.

Further Analysis. The λ parameter in Eq. 1 (Sect. 2) controls the reliance on retrieval scores versus inter-document-similarity information. We study the effect of varying λ on the p@5-performance of one of our most effective methods, BagDupMNZ, in Fig. 1. We can see that for most values of λ, and for most tracks, BagDupMNZ yields performance that transcends that of each of the three fused runs, and that of the optimized baseline. (The main exception is with respect to run1 for trec9.) We can also see that for all tracks, using $\lambda = 0.9$ — which is the optimal λ for most tracks — yields performance that is better than that of CombMNZ, which does not utilize inter-document-similarities. (Recall that for $\lambda = 1$ BagDupMNZ amounts to CombMNZ.) These findings further attest to the merits of using inter-document-similarities for fusion.

5 Conclusion

We presented a novel approach to fusing document lists that were retrieved in response to a query. Our approach integrates inter-document-similarities with retrieval scores of documents using a graph-based approach. Empirical evaluation demonstrated the effectiveness of the suggested models.

Acknowledgments. We thank the reviewers for their comments. This paper is based upon work supported in part by Google's and IBM's faculty research awards. Any opinions, findings and conclusions or recommendations expressed are those of the authors and do not necessarily reflect those of the sponsors.

References

1. Croft, W.B.: Combining approaches to information retrieval. In: [33], ch. 1, pp. 1–36.
2. Croft, W.B., Thompson, R.H.: I^3R: A new approach to the design of document retrieval systems. Journal of the American Society for Information Science and Technology 38(6), 389–404 (1984)
3. Fox, E.A., Shaw, J.A.: Combination of multiple searches. In: Proceedings of TREC-2 (1994)
4. Callan, J.P., Lu, Z., Croft, W.B.: Searching distributed collections with inference networks. In: SIGIR, pp. 21–28 (1995)
5. Lee, J.H.: Analyses of multiple evidence combination. In: Proceedings of SIGIR, pp. 267–276 (1997)
6. Das-Gupta, P., Katzer, J.: A study of the overlap among document representations. In: SIGIR, pp. 106–114 (1983)
7. Griffiths, A., Luckhurst, H.C., Willett, P.: Using interdocument similarity information in document retrieval systems. Journal of the American Society for Information Science (JASIS) 37(1), 3–11 (1986)
8. Chowdhury, A., Frieder, O., Grossman, D.A., McCabe, M.C.: Analyses of multiple-evidence combinations for retrieval strategies. In: Proceedings of SIGIR, pp. 394–395 (2001), poster
9. Soboroff, I., Nicholas, C.K., Cahan, P.: Ranking retrieval systems without relevance judgments. In: Proceedings of SIGIR, pp. 66–73 (2001)
10. Beitzel, S.M., Jensen, E.C., Chowdhury, A., Frieder, O., Grossman, D.A., Goharian, N.: Disproving the fusion hypothesis: An analysis of data fusion via effective information retrieval strategies. In: Proceedings of SAC, pp. 823–827 (2003)

11. van Rijsbergen, C.J.: Information Retrieval, 2nd edn., Butterworths (1979)
12. Kurland, O., Lee, L.: PageRank without hyperlinks: Structural re-ranking using links induced by language models. In: Proceedings of SIGIR, pp. 306–313 (2005)
13. Kurland, O.: Inter-document similarities, language models, and ad hoc retrieval, PhD thesis. Cornell University (2006)
14. Diaz, F.: Regularizing ad hoc retrieval scores. In: Proceedings of CIKM, pp. 672–679 (2005)
15. Pinski, G., Narin, F.: Citation influence for journal aggregates of scientific publications: Theory, with application to the literature of physics. Information Processing and Management 12, 297–312 (1976)
16. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: Proceedings of the 7th International World Wide Web Conference, pp. 107–117 (1998)
17. Golub, G.H., Van Loan, C.F.: Matrix Computations, 3rd edn. Johns Hopkins University Press, Baltimore (1996)
18. Aslam, J.A., Montague, M.: Models for metasearch. In: Proceedings of SIGIR, pp. 276–284 (2001)
19. Montague, M., Aslam, J.A.: Condorcet fusion for improved retrieval. In: Proceedings of CIKM, pp. 538–548 (2002)
20. Zamir, O., Etzioni, O.: Web document clustering: a feasibility demonstration. In: Proceedings of SIGIR, pp. 46–54 (1998)
21. Craswell, N., Hawking, D., Thistlewaite, P.B.: Merging results from isolated search engines. In: Proceedings of the Australian Database Conference, pp. 189–200 (1999)
22. Beitzel, S.M., Jensen, E.C., Frieder, O., Chowdhury, A., Pass, G.: Surrogate scoring for improved metasearch precision. In: Proceedings of SIGIR, pp. 583–584 (2005)
23. Selvadurai, S.B.: Implementing a metasearch framework with content-directed result merging, Master's thesis. North Carolina State University (2007)
24. Daniłowicz, C., Baliński, J.: Document ranking based upon Markov chains. Information Processing and Management 41(4), 759–775 (2000)
25. Zhang, B., Li, H., Liu, Y., Ji, L., Xi, W., Fan, W., Chen, Z., Ma, W.Y.: Improving web search results using affinity graph. In: Proceedings of SIGIR, pp. 504–511 (2005)
26. Kurland, O., Lee, L.: Respect my authority! HITS without hyperlinks utilizing cluster-based language models. In: Proceedings of SIGIR, pp. 83–90 (2006)
27. Otterbacher, J., Erkan, G., Radev, D.R.: Using random walks for question-focused sentence retrieval. In: Proceedings of HLT/EMNLP, pp. 915–922 (2005)
28. Diaz, F.: A method for transferring retrieval scores between collections with non overlapping vocabularies. In: Proceedings of SIGIR, pp. 805–806 (2008) (poster)
29. Diaz, F.: Performance prediction using spatial autocorrelation. In: Proceedings of SIGIR, pp. 583–590 (2007)
30. Erkan, G., Radev, D.R.: LexPageRank: Prestige in multi-document text summarization. In: Proceedings of EMNLP, pp. 365–371 (2004), poster
31. Mihalcea, R., Tarau, P.: TextRank: Bringing order into texts. In: Proceedings of EMNLP, pp. 404–411 (2004), poster
32. Zhai, C., Lafferty, J.D.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: Proceedings of SIGIR, pp. 334–342 (2001)
33. Croft, W.B. (ed.): Advances in Information Retrieval: Recent Research from the Center for Intelligent Information Retrieval. The Kluwer International Series on Information Retrieval, vol. 7. Kluwer, Dordrecht (2000)

A Quantum-Based Model for Interactive Information Retrieval[⋆]

Benjamin Piwowarski and Mounia Lalmas

University of Glasgow, Department of Computing Science,
Glasgow G12 8QQ, UK

Abstract. Even the best information retrieval model cannot always identify the most useful answers to a user query. This is in particular the case with web search systems, where it is known that users tend to minimise their effort to access relevant information. It is, however, believed that the interaction between users and a retrieval system, such as a web search engine, can be exploited to provide better answers to users. Interactive Information Retrieval (IR) systems, in which users access information through a series of interactions with the search system, are concerned with building models for IR, where interaction plays a central role. In this paper, we propose a general framework for interactive IR that is able to capture the full interaction process in a principled way. Our approach relies upon a generalisation of the probability framework of quantum physics.

1 Introduction

In less than twenty years, search engines on the Web have revolutionised the way people search for information. The speed with which one can obtain an answer to a keyword-based query on the Web is fostering interaction between search engines and their users. Helping users to reach relevant material faster will most likely make use of such rich interaction. Another key to future search systems is the context that further defines the search, whether it be external (e.g. time of the day, location) or internal (e.g. the interests of the user).

Putting aside the problem of evaluating such contextual and interactive search, building models able to explicitly take into account both is of importance, especially since Information Retrieval (IR) models seem to have reached maturity and there is an obvious need to go beyond current state-of-the-art [2].

There are many reasons why we cannot assume that users will provide enough information to state an unambiguous Information Need (IN), such as a TREC topic description. First, users do not always know how to express their IN and they sometimes have only a vague knowledge of what they are looking for. Second, users

[⋆] An extended version of this paper [1] contains the discussion about related works, a tutorial section about the relationship between quantum and classical probabilities, and an example of how the quantum formalism can be used to extend the Rocchio algorithm.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 224–231, 2009.

knowledge and interests might evolve during the search, thereby modifying their IN. Therefore, it is important that implicit contextual and interaction "information" become integrated directly into IR models *and* experiments [3].

Beside standard relevance feedback models like the Rocchio algorithm [4] or the Okapi model [5], some recent works have attempted to capture context [6] or interaction [7]. However, there is not yet a principled framework that combines both, and that, equally importantly, tries to capture the different forms of possible interactions, namely, query (re)formulation, clicks, navigation. Those tasks are all performed frequently in web searches.

In this paper, we present a framework for interactive and contextual IR. We view search as a process with two different dynamics: (P1) The system tries to capture the user IN while (P2) the user cognitive state, and hence the user IN, is evolving and changing [8]. While the former could be modeled by standard probabilistic models, we claim that the latter can be better modeled by the generalisation of probability theory that has been developed in quantum physics. Moreover, the strong geometric component of the quantum probability framework is particularly important since standard IR models rely on vector spaces and on (some variants of) the cosine similarity [9]. We show how the quantum formalism generalises these latter models (Section 3.1). In particular, we believe that one strength of the geometric models in IR is that they are intuitive. Adding a probabilistic view on this geometry opens the door for new and potentially more powerful IR models.

This paper describes how the quantum probability formalism could be used to build an interactive IR framework.

2 An Information Need Space

Our working hypothesis is that a *pure*, in the sense that we know exactly what the user is looking for, user IN can be represented as a system in quantum physics, i.e. as a unit vector in a Hilbert space[1], and that this state evolves while the user is interacting with the system.

According to the quantum probability formalism, this (IN) vector generates a probability distribution over the different subspaces of the Hilbert space. We make the hypothesis that among other possible uses, such subspaces can be related to the relevance of documents, therefore enabling the computation of a relevance score for a document, and to user interactions (like typing a query or clicking on a document), making it possible to exploit them.

From a geometric perspective, using subspaces to describe "regions" of INs has been (sometimes implicitly) studied and motivated in some works relying on a vector space representation [10,11,12]. Using those IN "regions", the search process would be modelled as follows. At the very beginning of the search process, the user IN is underspecified and is a mixture of *all* possible pure INs. That is, without any information about the user, we can only know that the user is in

[1] In brief, an inner product vector space defined over the complex field, see [9] for a formal definition.

one of all the possible IN states with a probability that depends e.g. on how popular this IN is.

We believe that using an IN space can model interactive IR since users change their point of view during a search, and relevance, contrarily to topicality, is expected to evolve within a search session [8,13]. More specifically, we can identify two different types of dynamics within the search process: (P1) The IN becomes increasingly specific *from a system point of view*, e.g. when a user types some keywords or clicks on some documents, i.e. the uncertainty is reduced; and (P2) The IN changes *from a user point of view*. The IN can become more specific as the user reads some documents, or it can slightly drift as user interests do.

Whereas the first process can be easily described within a standard probabilistic framework (we restrict the IN to subspaces of the whole space), the latter would benefit from a quantum probability formalism as the INs can drift from two overlapping subspaces. We posit that the classical probabilistic framework would address the uncertainty of the system view over the retrieval process (P1) whereas the quantum probability framework addresses the changes of the user internal state (P2). As the quantum probability framework is a generalisation of the probabilistic one, we can use the same representation and evolution operators to model both processes.

3 A Quantum View

Quantum probability can be thought of as an extension of classical probability theory, and relies on linear algebra in Hilbert spaces. The equivalent of a logical proposition or event A is a subspace or equivalently [9] its associated projector O_A which is called a *yes/no observable*.

All the information about the probability distribution is contained into a *density operator* ρ, and it can be shown that for *any* probability distribution over a Hilbert space there exists a corresponding density operator [9, p. 81]. A density operator ρ can be written as a mixture of projectors $\rho = \sum_O \Pr(O)\, O$ where the sum ranges over projectors O and $\Pr(O)$ sum up to 1. Note that a pure state is defined as a density which is equal to a one-dimensional projector. Denoting tr the trace operator, the probability of the event O_A for the density operator ρ is then given by

$$\Pr_\rho(O_A) \doteq \mathrm{tr}(\rho O_A) \tag{1}$$

From a practical point of view, the above description of standard probabilities with Hilbert spaces unlocks the potential of defining probabilities through geometric relationships, and permits a generalisation to a non standard probability formalism, which we describe in the next section. We posit that at this level, we are able to model the first component of the search process, which corresponds to finding the right *subspace* of the IN, i.e. in classical terms to find the subset of the IN sample space. However, it is intuitive to think that INs are not mutually exclusive. We make the hypothesis that such a non-exclusiveness is captured by the geometry of IN space, and this can be modelled within a quantum probability formalism.

3.1 Superposition, Mixtures and Information Needs

We introduce the notion of superposition and mixture, and relate them to their use in our model of interactive IR. Said shortly, superposition relates to an ontologic uncertainty (the system state is perfectly known, but some events are true *only* with a given probability) whereas mixture relates to standard probabilistic uncertainty (the system is in one of the states with a given probability). Superposition is a salient characteristic of quantum probabilities and is important since it gives us a way to represent geometrically new INs while the quantum probability framework ensures we can still compute probabilities for the new INs. Mixture and superposition gives us more flexibility in the way we can represent our current state of knowledge of an IN.

Let us illustrate this with an example. Suppose that $\omega_T = \begin{pmatrix} 1 \; 0 \end{pmatrix}^\top$ and $\omega_L = \begin{pmatrix} 0 \; 1 \end{pmatrix}^\top$ form a basis of the IN space ($\top$ denotes the transpose of a matrix). Suppose the (projector associated to the) former represents the IN of a user looking for information about tigers (T) and the latter about lions (L). In order to represent a user looking for a tigron (the offspring of a tiger and a lion), we assume that this can be represented by (the projector associated to) the vector $\omega_{TL} = \frac{1}{\sqrt{2}} (\omega_T + \omega_L)$ which is a *superposition* of two INs, where the $\frac{1}{\sqrt{2}}$ factor ensures ω_{TL} norm is one. This is a strong assumption which we will study when experimenting with the framework. Aerts and Gabora [14] worked on how to combine concepts in a (quantum) vector space, but use spaces of increasing dimensionality to do so (through the use of a tensor product). As a final remark on superposition of INs, we would like to note that complex numbers could be used to combine INs, e.g. to distinguish tigrons (the tiger is the father) from ligers (the lion is the father), and that superposition is not restricted to topicality. For instance, assuming that we know how to represent a user searching for a paragraph and a user searching for a chapter, we could imagine representing a user looking for a paragraph as a superposition of both.

The superposed IN ω_{TL} is quite different to the IN of a user who is equally interested by tigers or lions. The latter would be represented as a *mixture* of the INs ω_T and ω_L. Formally, this IN would be associated with a density operator$\rho_{T\vee L} = \frac{1}{2} (\rho_T + \rho_L)$ where ρ_L and ρ_T are respectively the projectors associated with ω_T and ω_L, e.g. $\rho_T = \omega_T \omega_T^\top$. The density operator $\rho_{T\vee L}$ is to be interpreted by saying that with probability one half the IN is about tigers (or equivalently about lions).

We can see also the difference if we represent the densities by their matrices in the (ω_T, ω_L) basis. We have the mixture of IN $\rho_{T\vee L} = \frac{1}{2} \begin{pmatrix} 1 \; 0 \\ 0 \; 1 \end{pmatrix}$ which is different from the pure IN $\rho_{TL} = \frac{1}{2} \begin{pmatrix} 1 \; 1 \\ 1 \; 1 \end{pmatrix}$. An important observation is that these different densities imply different probabilities. Let us suppose that the relevance of a document corresponds to a yes/no observable, and that the relevance of a document about lions (respectively tigers, tigrons) are represented by the projectors (yes/no observables) O_L, O_T and O_{TL} associated with the subspaces generated by ω_T, ω_L and ω_{TL}, respectively. For example, $O_T = \omega_T \omega_T^\top$. According to Eq. (1), we can compute the probability of relevance of the different documents, which gives:

$$\Pr_{\rho_{TL}}(O_L) = \Pr_{\rho_{T\vee L}}(O_L) {=} \tfrac{1}{2} \text{ and } \Pr_{\rho_{TL}}(O_{TL}) = 1 \neq \Pr_{\rho_{T\vee L}}(O_{TL}) = \tfrac{1}{2}$$

Interestingly, we cannot distinguish the probability of relevance of the document about lions when the IN is about either tigers and lions or about tigrons (two first probabilities) but there are two reasons for this: In the former, the probability $\frac{1}{2}$ is caused by the discrepancy between the IN and the document, whereas in the second case the probability is due to the fact that the document only covers a part of the information need. Next, thanks to the quantum formalism the probabilities for the same INs are different when we evaluate the relevance of the document about tigrons (two last probabilities). We thus benefit from a two-dimensional space to distinguish different INs that would be expressed similarly in a standard vector space model. One consequence is that if we search for a set of documents that satisfy T or L, we would have two different types of documents (about tigers and lions, assuming each document covers one IN only) whereas one document would satisfy TL.

Mixtures are also useful to represent the IN density operator ρ_0 at the very beginning of the information retrieval process, as we do not know which state the user is in. We would define the initial IN density operator as $\rho_0 = \sum_i \Pr_\mathrm{i} \mathbb{P}_\mathrm{i}$ where i ranges over all the possible pure information needs $\mathbb{P}_\mathrm{i}$ and $\Pr_i$ is the probability that a random user would have the IN i when starting a search. Using the mixture is also motivated by the fact that we deal with classical undeterminism, i.e. we know the user is in a given state but we do not know which. The mixture can also be thought as a set of vectors describing all the possible INs, each vector being associated with a probability. This representation is particularly useful in the next section where we show how this initial IN ρ_0 is transformed through interactions.

3.2 Measurement and Interaction

Beside differentiating mixture and superpositions, the quantum formalism has also consequences for computing a conditional probability. These consequences are linked to the way a measurement is performed in quantum physics. We use measurement to model interaction and describe in this section both how the measurement modifies the density operatorρ and how we link measurement to the different interactions.

For simplicity, we now use O_A to denote the related yes/no observable, subspace or projector. Since there is a one-to-one correspondence between them [9], they can be used to denote the same thing albeit in a different context. Given a system density operator ρ, if we observe O_A, the new density operator denoted $\rho \rhd O_A$ is defined by

$$\rho \rhd O_A = O_A \rho O_A / \mathrm{tr}\,(\rho O_A) \tag{2}$$

This amounts to restricting ρ to the subspace defined by O_A and ensuring that $\rho \rhd O_A$ is still a density operator. The effect of the restriction is to project every IN of the mixture ρ onto the subspace defined by O_A (with some renormalisation to ensure the probabilities still sum up to 1). One can readily verify that the probability of O_A with respect to $\rho \rhd O_A$ is 1. It means that when A has just be measured, we know it is true at least until further interaction (or in general, evolution) modifies the density operator. Measurement can be thought as a generalisation of conditionalisation, as we can compute the conditional probability of O_A given O_B, or more precisely of measuring O_A knowing that we have measured O_B, as $\Pr_\rho(O_B|O_A) = P_{\rho \rhd O_A}(O_B)$.

In quantum theory, the order of the measurements is important, since in general the densities $\rho \rhd O_A \rhd O_B$ (applying two times the Eq. (2), for O_A and then for O_B) and $\rho \rhd O_B \rhd O_A$ are different. It is a desirable property whenever subsequent measurements of a system should yield different results, which is the case in interactive IR: The sequence of interactions represents the evolution of the user, and should be taken into account. A user drifting from an IN (e.g. hotels in Barcelona) to another (e.g. museum in Barcelona) is not the same as the reverse, which illustrates the adequacy of the quantum formalism to handle such drifts. This is illustrated by Figure 1, where visually it can be seen that measuring O_B (hotels) then O_C (museums) is different from the reverse, since in the first case the IN vectors will lie in the subspace C whereas they would lie in B in the other case.

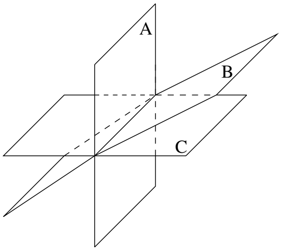


Fig. 1. Three two-dimensional subspaces (A, B, C) in a three dimensional space

Starting with the initial density operator ρ_0 (section 3.1), we make the assumption that each implicit or explicit interaction between the IR system and the user corresponds to a measurement, i.e. that every interaction is associated with a yes/no observable O. After the interaction, we can recompute the IN density operator using Eq. (2). For example, a user whose internal context is associated as O_{user}, who asked a query associated with O_{q_1} and deemed a document relevant (associated with O_{d_1}), would be represented by a density operator $\rho_0 \rhd O_{\text{user}} \rhd O_{q_1} \rhd O_{d_1}$. Among other users, this density operator can be used to predict the relevance of other documents.

3.2.1 Mapping Interactions to Observables

In order to map interactions to observables, we restrict to the topical relevance and assume a vector space where dimensions are associated with terms. How to deal with more relevance dimensions is left for future work. We also assume we know how to compute the initial density operator ρ_0 – which could be approximated using the document representation described next.

Giving the current IN density operator ρ_t, we can compute the probability of relevance $\Pr_{\rho_t}(O_d)$ of a document d , provided O_d is the observable associated with the relevance of document d. To build such an observable, and as a first approximation, we can suppose that each paragraph p corresponds to exactly one IN ω_p, and hence that its representation is a one dimensional subspace. It is then possible to compute the subspace spanned by the vectors $\{\omega_p\}$ corresponding to the different paragraphs, and use this subspace to represent the document relevance. When a user deems a document relevant, we could use the same representation to update the current IN ρ_k. In that case, we would have the new IN density operator $\rho_{t+1} = \rho_t \triangleright O_d$.

The first possible type of interaction would be the (re)formulation of a query by a user. We would associate to a given query a subspace/observable O_q, and update the current probability density operator ρ_t to $\rho_{t+1} = \rho_t \triangleright O_q$. A representation of the query could for example be computed through pseudo-relevance feedback provided we know how to represent the documents: The subspace associated with O_q would then be the subspace spanned by the observables representing the top-ranked documents (by a standard IR algorithm). For example, in Figure 1, if A and B correspond to two different top-ranked documents for a given query, then O_q would correspond to the whole three dimensional space (i.e. the join of subspaces A and B). Another way to compute the query observable O_q, without relying on an external model, would be the union of the subspaces representing the paragraphs where each term of the query appears.

Here, we give one illustration of the usefulness of the quantum formalism for an interactive IR framework. The query observable O_q (or the document observable O_d) can be used to detect if a user's change of mind is too important to be a simple drift, an important feature an interactive IR system should have [13]. Within the quantum framework, we use the same geometric representation to both update the density operator knowing an event and to compute the probability of this event. Indeed, when at time t the user types a new query q', we can compute the probability of the query according to the current IN density operator ρ_t, i.e. compute $\Pr_{\rho_k}(O_{q'})$. Based on this value, our IR system would decide that the user switched to a new IN, and react accordingly.

4 Conclusion

We proposed a new interactive IR framework, which exploits the strong connection between geometry and probabilities present in the quantum probability formalism. Our framework allows for a principled and geometric mapping of user interactions into an IR model. In particular, we show how to handle click/ relevance feedback and query reformulation. How to use the latter information has not been explored in IR so far, beside providing query recommendation. Other forms of interaction (e.g. navigation) would fit our framework, through the definition of associated subspaces. Beside measurement, the quantum framework is powerful enough to provide other types of evolution of the IN density operator. This would provide a way to predict how a user might evolve, e.g. in order to predict that users looking for hotels might look for museums in a town.

Acknowledgments. This research was supported by an Engineering and Physical Sciences Research Council grant (Grant Number EP/F015984/2). Mounia Lalmas is currently funded by Microsoft Research/Royal Academy of Engineering.

References

1. Piwowarski, B., Lalmas, M.: A Quantum-based Model for Interactive Information Retrieval (extended version). ArXiv e-prints (0906.4026) (2009)
2. Sparck Jones, K.: What's the value of TREC: is there a gap to jump or a chasm to bridge? SIGIR Forum. 40(1), 10–20 (2006)
3. Ingwersen, P., Järvelin, K.: The Turn: Integration of Information Seeking and Retrieval in Context. The Information Retrieval Series. Springer, USA (2005)
4. Rocchio, J.J.: Relevance feedback in information retrieval. In: Salton, G. (ed.) The SMART retrieval system: experiments in automatic document processing, pp. 313–323. Prentice-Hall, Englewood Cliffs (1971)
5. Walker, S., Robertson, S.E.: Okapi/keenbow at TREC-8. In: Voorhees, E.M., Harman, D.K. (eds.) NIST Special Publication 500-246: The Eighth Text REtrieval Conference (TREC-8), Gaithersburg, Maryland, USA (November 1999)
6. Melucci, M.: A basis for information retrieval in context. ACM Transactions on Information Systems 26(3), 1–41 (2008)
7. Shen, X., Tan, B., Zhai, C.: Implicit user modeling for personalized search. In: Proceedings of the 14th ACM international conference on Information and knowledge management, Bremen, Germany, pp. 824–831. ACM, New York (2005)
8. Xu, Y.: The dynamics of interactive information retrieval behavior, part i: An activity theory perspective. Journal of the American Society for Information Science and Technology 58(7), 958–970 (2007)
9. van Rijsbergen, C.J.: The Geometry of Information Retrieval. Cambridge University Press, New York (2004)
10. Wang, X., Fang, H., Zhai, C.: A study of methods for negative relevance feedback. In: Myaeng, S.H., Oard, D.W., Sebastiani, F., Chua, T.S., Leong, M.K. (eds.) Proceedings of the 31st Annual International ACM SIGIR, pp. 219–226. ACM, New York (2008)
11. Widdows, D.: Orthogonal negation in vector spaces for modelling word-meanings and document retrieval. In: Proceedings of the 41st ACL conference, Association for Computational Linguistics, Morristown, NJ, USA, pp. 136–143 (2003)
12. Zuccon, G., Azzopardi, L., van Rijsbergen, C.J.: Semantic spaces: Measuring the distance between different subspaces. In: Bruza, P., Sofge, D., Lawless, W., van Rijsbergen, C.J., Klusch, M. (eds.) Proceedings of the Third Quantum Interaction Symposium. LNCS (LNAI), vol. 5494. Springer, Heidelberg (2009)
13. Xie, H.I.: Shifts of interactive intentions and information-seeking strategies in interactive information retrieval. Journal of the American Society for Information Science 51(9), 841–857 (2000)
14. Aerts, D., Gabora, L.: A theory of concepts and their combinations II: A Hilbert space representation. Kybernetes 34 (2005)

The Quantum Probability Ranking Principle for Information Retrieval

Guido Zuccon*, Leif A. Azzopardi, and Keith van Rijsbergen

Department of Computing Science
University of Glasgow
Scotland, UK
{guido,leif,keith}@dcs.gla.ac.uk

Abstract. While the Probability Ranking Principle for Information Retrieval provides the basis for formal models, it makes a very strong assumption regarding the dependence between documents. However, it has been observed that in real situations this assumption does not always hold. In this paper we propose a reformulation of the Probability Ranking Principle based on quantum theory. Quantum probability theory naturally includes interference effects between events. We posit that this interference captures the dependency between the judgement of document relevance. The outcome is a more sophisticated principle, the Quantum Probability Ranking Principle, that provides a more sensitive ranking which caters for interference/dependence between documents' relevance.

1 Introduction

The core task of Information Retrieval (IR) is to retrieve a set of documents satisfying a user's information need [6]. A key paradigm in IR [4] employs formal theories to estimate the probability of relevance of a document given a user's information need. In order to achieve an optimal retrieval performance, documents retrieved by the IR system are ranked in accordance to the Probability Ranking Principle (PRP) [5]. This posits that the system should rank documents in decreasing order of their probability of being relevant to the user's information need. Among others, one of the most controversial assumption made by the PRP is that the relevance of a document to an information need does not depend on other documents (*independent relevance* assumption). However, in real search situations the judgement of relevance made by the user about a document is influenced by the documents he previously examined through the search process [2]. Moreover, it has been shown that the utility of a document might become void if the user has already obtained the same information. This "interference" is due to several factors such as changes in information need, or information overlap among documents, or contrary information and is not

* Supported by EPSRC Grant number EP/F014384/. The authors are grateful to A. Huertas, D. Elliot, M. Lalmas, B. Piwowarski for their insightful comments.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 232–240, 2009.

accounted for by the PRP as relevance of a document judgements is assumed independent from other documents.

In this paper, we model the PRP using quantum probability. The formulation of the PRP based on quantum probability naturally encodes quantum interference, which can be interpreted as modeling dependent relevance, thus overcoming the independent relevance assumption made by the original PRP.

The remainder of the paper follows. In Section 2 we present the double slit experiment, drawing a metaphor between IR and Physics. The classical PRP will be framed in the proposed analogy (Section 3), while arising of interferences in the experiment will be the stimulus towards a ranking principle which accounts for interference, the QPRP. In Section 4 we discuss a possible interpretation of the interference term in IR. The paper concludes stating the contribution of this study and lines of future work (Section 5).

2 The Double Slit Experiment

In this section we introduce quantum probabilities and the quantum interference effect. Quantum interference is of major importance in our approach. To illustrate the difference between Kolmogorovian and quantum probabilities, we present a simple physical test, the double slit experiment [3], which describes arising the of interference between the probabilities of two events. The double slit experiment consists of shooting a physical particle (i.e. an electron, a photon, etc.) towards a screen with two slits, named A and B (Fig. 1(a)). Once the particle passes through one of the slits, it hits a detector panel, positioned behind the screen, in a particular location x with probability $p_{AB}(x)$.

By closing slit B, it is possible to measure the probability of the particle being detected in position x passing through A, namely $p_A(x)$. Similarly, by closing just slit A, we can measure $p_B(x)$. We call ϕ_A the (complex) probability amplitude associated to the events of passing through A when B is closed and being detected at x, and vice-versa for ϕ_B. The following equations state the relationship between probability and probability amplitudes: $p_A(x) = |\phi_A(x)|^2$; $p_B(x) = |\phi_B(x)|^2$. Intuitively[1], we would expect that the probability of the particle being detected at x when both slits are open is the sum of the probability of passing through A and being detected at x, $p_A(x)$, and the probability of passing through B and hit the detector panel in x, $p_B(x)$. Formally,

$$p_{AB}(x) = p_A(x) + p_B(x) = |\phi_A(x)|^2 + |\phi_B(x)|^2 \qquad (1)$$

We refer to this case with the adjective *classical* meaning that no quantum phenomena would be observed. However, experimentally it has been noted that $p_{AB}(x) \neq p_A(x) + p_B(x)$, i.e. the probability of the particle being detected at x when both slits are open *is not* the sum of the probability with just slit A open plus that with just slit B open. Actually, the probability distribution that can be obtained measuring $p_{AB}(x)$ across the whole detection panel presents an

[1] And applying the Kolmogorovian law of total probability.

interference pattern akin to waves that would pass through both slits and hit the detector panel. Thus, representing with $\phi_{AB}(x)$ the (complex) probability amplitude of a particle being measured at position x after passing through either slit A or B, it is possible to state that $\phi_{AB}(x)$ is the sum of the probability *amplitude* associated to the event of opening just slit A plus the counterpart event of having open just slit B. In other words, $\phi_{AB}(x) = \phi_A(x) + \phi_B(x)$, and the probability of such event is $p_{AB}(x) = |\phi_{AB}(x)|^2$. The application of the previous relationships involving probabilities amplitudes results in

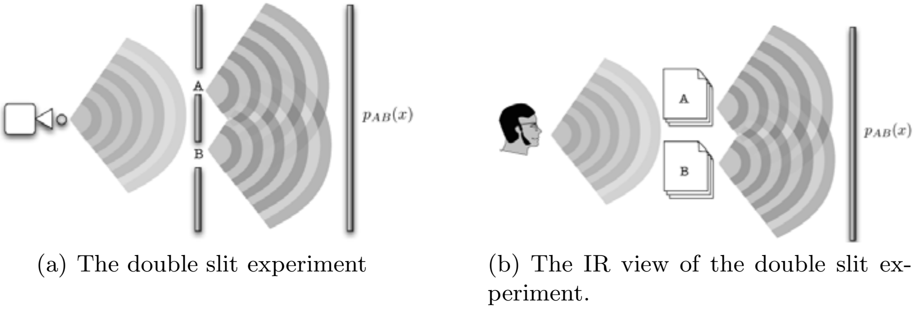
$$\begin{aligned} p_{AB}(x) &= |\phi_A(x)|^2 + |\phi_B(x)|^2 + (\phi_A(x)^*\phi_B(x) + \phi_A(x)\phi_B(x)^*) \\ &= p_A(x) + p_B(x) + I_{AB}(x) \end{aligned} \tag{2}$$

The term $I_{AB}(x)$ in Eq. 2 represents quantum *interference* between the events associated to $p_A(x)$ and $p_B(x)$ and is modulated by the phase difference between the correspondent amplitudes.

In summary, the conventional Kolmogorovian rule for addition of probabilities of alternatives, Eq. 1, is violated in the double slit experiment. When the event can occur in several alternative ways, the probability amplitude of the event, $\phi_{AB}(x)$, is the sum of the probability amplitude (the absolute square of a complex quantity) for each alternative considered separately. In the case of quantum probabilities, Eq. 1 is re-written with the addition of a perturbation term (shown in Eq. 2). The interpretation and the behavior of the interference term will be discussed later (Section 4); in the following we devise an analogy between the double slit experiment and the IR ranking process.

3 The Analogy

In the following, we discuss (i) the classical PRP in terms of its decision theory derivation, adopting the analogy of the double slit experiment without interference effects, (ii) the case in which interference effects arise, and (iii) the derivation from the analogy of the new ranking principle.



(a) The double slit experiment

(b) The IR view of the double slit experiment.

Fig. 1. Schematic representation of the analogy between the double slit experiment and the IR ranking problem

We propose an analogy between the double slit experiment and the IR situation. In our analogy, the particle is associated with the user and his information need, while each slit represents a document. The event of passing from the left of the screen to the right (through a slit) is seen as the action of examining the ranking of documents, e.g. read the associated snippets or the documents themselves. Measuring at x means assessing the satisfaction of the user given the presented ranking of documents, or more concretely the decision of the user to stop his search (event x, the user is fully satisfied) or continue searching ($\bar{x}$, he is not completely satisfied by the documents presented). Thus, being detected with probability $p_{AB}(x)$ at position x on the panel means choosing to stop the search with probability $p_{AB}(x)$ after being presented with documents A and B. This scenario is represented in Fig 1(b). The user is presented with two documents, A and B, and he has to decide whether to stop the search (event x, associated probability $p_{AB}(x)$) or to continue (event $\bar{x}$, probability $p_{AB}(\bar{x}) = 1 - p_{AB}(x)$).

Probability $p_{AB}(x)$ is influenced by the characteristics of slits (documents) A and B. Consider the case several experiments are ran varying the screen among a set of them, all having the same slit A but each of them being characterized by a different slit B: e.g. B_h is narrow while B_j is wide, B_k is close to A while B_l is farer apart from A. The set of all different slits B_i is identified by $\mathfrak{B}$ and in our analogy it represents the set of candidate documents to be ranked immediately after document A.

Following the analogy, maximizing the expected utility of the ranking of documents is seen as maximizing the probability $p_{AB_i}(x)$, i.e. the probability of stopping the search having seeing A and B_i and, in the case of the physical experiment, maximizing the probability of the particle hitting the detector panel at position x (stop the search) passing though one of the slits. The problem then concretizes in determine which configuration of slits AB_i with $B_i \in \mathfrak{B}$ exhibits maximal $p_{AB_i}(x)$.

The classical case. If the double slit experiment is modeled assuming no interference, i.e. the "classical" case, the maximum $p_{AB_i}(x)$ is obtained by the configuration of slits with maximal $p_{B_i}(x)$. In fact, the probability of being detected at x being passed through either A or B_i is given by Eq. 1, and thus imposing maximal $p_{AB_i}(x)$ is equivalent to maximize $p_A(x) + p_{B_i}(x)$. However, since $p_A(x)$ is constant among all screen's configuration, we obtain

$$\underset{x}{\operatorname{argmax}}\big(p_{AB_i}(x)\big) = \underset{x}{\operatorname{argmax}}\big(p_A(x) + p_{B_i}(x)\big) = \underset{x}{\operatorname{argmax}}\big(p_{B_i}(x)\big) \qquad (3)$$

In IR terms, given a fixed A (the document at first position of the ranking), the best document B_i to select among all the candidates $\mathfrak{B}$ is given by the document which maximizes $p_{AB_i}(x)$. In the classical case, $p_{AB_i}(x)$ is given by Eq. 1, and then maximizing it means choosing the document B_i with maximal $p_{B_i}(x)$, the document among the candidates $\mathfrak{B}$ with maximal probability of inducing the user to stop the search, i.e. probability of relevance.

In summary, maximizing the outcome of the measurement of a particles system passing through slits by choosing which pair of slits to use is analogous to choose which document to rank next, given a set of possible documents to rank.

In absence of interference, the optimal rank suggested by the analogy with the double slit experiment is in accordance with the PRP: the slit B_i that should be used in order to maximize $p_{AB_i}(x)$ is the one for which $p_{B_i}(x)$ is maximal.

The quantum case. In the following we examine the situation where quantum phenomena appears in the double slit experiment and from this we abstract and derive a formulation of the PRP based on quantum probabilities. Maintaining the same analogy exploited previously, in presence of interference the probability $p_{AB}(x)$ is governed by Eq. 2. The probability of the particle being measured at position x in the detector panel is given by the sum of the probability of the particle being measured at x and passing either through A (term $p_A(x)$) or B (term $p_B(x)$), and a third term, the interference between the phases of the probability amplitudes associated to the mutually exclusive events of passing through Y ($Y = A, B$) and being measured at x.

We suppose to have at our disposal a set of screens with a fixed slit A and different implementation of a second slit B_i. We aim to select the configuration of slits A and $B_i \in \mathfrak{B}$ which maximize probability $p_{AB_i}(x)$, representing the probability of finding a particle at position x on the detection panel after it passed either by slit A or B_i, analogous in the instituted metaphor to the probability of a user deciding to stop his search (because satisfied of the results obtained) after having examined either document A or B_i.

In presence of interference, $p_{AB_i}(x) = p_A(x) + p_{B_i}(x) + I_{AB_i}(x)$ leading to

$$\begin{aligned}\underset{x}{\operatorname{argmax}}\big(p_{AB_i}(x)\big) &= \underset{x}{\operatorname{argmax}}\big(p_A(x) + p_{B_i}(x) + I_{AB_i}(x)\big)\\ &= \underset{x}{\operatorname{argmax}}\big(p_{B_i}(x) + I_{AB_i}(x)\big) \qquad (4)\end{aligned}$$

since $p_A(x)$ is constant among all the available screens. Allowing quantum interference, the maximum $p_{AB_i}(x)$ is reached when the sum $p_{B_i}(x) + I_{AB_i}(x)$ is maximal. The choice of the optimal screen among the possible screens with pairs of slits (A, B_i), $B_i \in \mathfrak{B}$ is not the same as in the classical case (the pair for which $p_{B_i}(x)$ is maximal) but depends upon $p_{B_i}(x)$ and the interference between A and B_i, $I_{AB_i}(x)$.

Deriving the Quantum PRP. The analogy suggests that the best choice for the document to rank after A is not the one for which $p_{B_i}(x)$ is maximal, i.e. the probability of relevance is maximal among the possible candidates $\mathfrak{B}$. Optimal rank would be produced when taking into account also the interference term. The probability of a document Y inducing the user to stop his search because his information need has been satisfied by the document is proportional to the probability of relevance to the information need of the document itself: $p_Y(x) \propto P(R|Y, q)$. We define $u(x)$ and $u(\bar{x})$ as the utility of retrieving a document which induces the user to stop his search and the utility of retrieving a document which does not induce the user to stop his search, respectively. We can safely assume $u(x) > u(\bar{x})$, setting for convenience $\big(u(x) - u(\bar{x})\big) = U$. Then, the expected utility in presence of interference can be written as:

$$\mathfrak{U} = p_A(x)U + p_Y(x)U + I_{AY}(x)U + u(\bar{x}) \qquad (5)$$

The maximum value of expected utility is reached for the configuration which exhibits the maximum $p_Y(x) + I_{AY}(x)$, in fact $\mathrm{argmax}(\mathfrak{U}) = \mathrm{argmax}\big(p_Y(x) + I_{AY}(x)\big)$. When evaluating which is the optimal document to rank after A not only probability $p_Y(x)$ has to be taken into account, but also the probability of interference between the two documents affects the expected utility. Thus if dealing with quantum probabilities, document B should be ranked immediately after A and before any other document C if and only if

$$\begin{aligned} u(x)p_{AB}(x) + u(\bar{x})p_{AB}(\bar{x}) &\geq u(x)p_{AC}(x) + u(\bar{x})p_{AC}(\bar{x}) \\ \Leftrightarrow \boxed{p_B(x) + I_{AB}} &\geq \boxed{p_C(x) + I_{AC}} \end{aligned} \tag{6}$$

that is, B is the document belonging to $\mathfrak{B} = \mathfrak{Y} \setminus \{A\}$ for which $p_B(x) + I_{AB}$ is maximal. The statement of the Quantum PRP follows:

> *The quantum probability ranking principle (QPRP):* in order to maximize the effectiveness of an IR system, document B should be ranked after the set $\mathfrak{A}$ of documents already ranked and before any other document C in the list returned to the user who submitted the query if and only if $p_B(x) + I_{\mathfrak{A}B} \geq p_C(x) + I_{\mathfrak{A}C}$, where $I_{\mathfrak{A}Y}$ is the sum of all the interference terms associated to each pair of documents Y and $X \in \mathfrak{A}$.

Note that both the classical PRP and its quantum counterpart posit that the document at the first position of the ranking is the one with highest probability of relevance given the information need, since this is the document associated with the highest expected utility.

4 Discussion

In the quantum version of the PRP, the interference probability has a major role; but, **what is its interpretation?** We hypothesize that in IR interference occurs in the ranking between documents (or representations of them) at the relevance level. For example, [1] and [7] showed that the user is more likely to be satisfied by documents addressing his information need in different aspects than documents with the same content. Then, it might be sensible to model documents expressing diverse information as having higher degree of interference than documents that are similar. For the same reason, documents containing novel information might highly interfere with documents ranked in previous positions. Even contrary information might be captured by the interference term: documents containing content contrary to the one presented at the previous rank position might trigger a revision of user's beliefs about the topic. In summary, interference might model dependencies in documents' relevance judgements: the QPRP suggests that documents ranked until position $n-1$ interfere with the degree of relevance of the document ranked at position n. The classical PRP does not take into account dependent relevance of documents. Conversely, due to the presence of the interference term, the quantum ranking principle models dependent relevance and might be suited to address novelty/diversity in the documents ranking.

In what ways does the QPRP differ from the PRP? Both the classic PRP and its quantum counterpart posit that the document at the first position of the ranking is the one with highest probability of relevance given the information need, e.g. document A. The PRP ranks the documents that are left in decreasing order of relevance, while the QPRP postulates interference has to be taken into account. In the PRP the decision to rank a document in a particular position is not determine by the documents retrieved at previous ranks but only upon the relevance score assigned to other documents candidate to be ranked (i.e. independent relevance). Conversely, the interference term in the QPRP depends upon the documents ranked at previous positions. This means, the optimal order of documents under the PRP is different to that of the QPRP, and such difference is influenced by the interference term. **How does the interference term influence ranking of documents?** Consider Table 1. Assume $p_B(x)$ is greater than $p_C(x)$; then the PRP ranks B before C. However, from Eq. 6 the quantum PRP behaves in the same way (rank B before C) if and only if the difference between the probabilities associated to the single documents $(p_B(x) - p_C(x))$ is greater than the difference between their interference terms $(I_{AC}(x) - I_{AB}(x))$. Conversely, if this is not the case (i.e. $p_B(x) - p_C(x) < I_{AC}(x) - I_{AB}(x)$), the QPRP imposes to rank C before B. Then document C is promoted above B because its interference with the document ranked at the previous position (A) is so high that it fills the gap given by $p_B(x) + I_{AB}(x) - p_C(x)$. We interpret then document C as a document carrying diverse and novel information related to the query with respect to document A, while document B's content is less novel or possibly not novel at all with respect to document A. Moreover, when B and C are equally probable to be relevant ($p_B(x) = p_C(x)$), the PRP ranks first either one of them. However, in the same situation, the QPRP favors B above C if and only if the probability of B interfering with A is greater than the one of the pair (A, C). It is a matter of empirical investigation to determine how many times the rankings provided by the classical PRP and by its quantum counterpart differ.

What governs the interference term? Recall that the probability associated to the interference is given by $I_{AB}(x) = 2\,|\phi_A(x)|\,|\phi_B(x)| \cos\theta_{AB} = 2\sqrt{p_A(x)p_B(x)}\cos\theta_{AB}$, where θ is the difference of the phases of $\phi_A(x)$ and $\phi_B(x)$. When $\cos\theta_{AB} > 0$, $I_{AB}(x)$ is called constructive interference; conversely, destructive interference is obtained when $\cos\theta_{AB} < 0$. The behavior of the probability of the interference is governed by the phase θ.

How does interference behave by varying θ? The phase actively affects the documents ranking. For example, when $p_B(x) = p_C(x)$, document B would be ranked above document C when $\cos\theta_{AB} > \cos\theta_{AC}$. In general, when $p_B(x) \geq p_C(x)$ the interference term is able to subvert the ordering suggested by the classical PRP (i.e. "rank B above C") if

$$\frac{p_B(x) - p_C(x)}{2\sqrt{p_A(x)}} < \sqrt{p_C(x)}\cos\theta_{AC} - \sqrt{p_B(x)}\cos\theta_{AB} \tag{7}$$

Table 1. When does B have to be ranked above C? A comparison between classical PRP and its quantum counterpart (QPRP)

	$p_B(x) > p_C(x)$	$p_B(x) = p_C(x)$
PRP	B before C	either
QPRP	B before C iff $p_B(x) - p_C(x) > I_{AC}(x) - I_{AB}(x)$	B before C iff $I_{AB}(x) > I_{AC}(x)$

How is θ computed in IR? While $p_A(x)$, $p_B(x)$, etc., are estimated from statistical feature of the document collection, the computation of the phase θ is still an open question and will be subject of further investigation. However, we suggest that θ could be approximated using the cosine similarity between documents. In particular, $\theta_{AB} \approx \arccos(\text{sim}\,(A, B)) + \pi$. Alternative strategies might relate θ to the information gain or cross entropy between documents.

In summary, interference occurs between documents at relevance level. While the classical version of the PRP does not provide optimal ranking in presence of interference, the quantum PRP copes with this situation, promoting documents that positively interfere at relevance level.

5 Conclusions

In this paper we exploit an analogy between the ranking problem in IR and the double slit experiment. The analogy introduces the presence of quantum interference between events. Taking into account the probability of interference, a new version of the Probability Ranking Principle, namely the Quantum PRP, has been proposed. We showed that the quantum version of the principle is a generalization of the classical PRP, and that it leads to optimal ranking solutions in presence of interference. In particular, it has been proposed that the interference term models the relationships between documents at the relevance level. Then, the document independency assumption needed for the classical PRP can be dropped in its quantum counterpart. In practice, the interference term is governed by the phase θ. The estimation of the phase in an effective way for IR is still an open issue; however, we have suggested possible avenues of research. To the best of our knowledge, our approach is the only that models dependent relevance in a principled way. It is interesting to investigate if other strategies which might violate the classical PRP, e.g. [1,7], uphold for the QPRP.

References

1. Chen, H., Karger, D.R.: Less is more: probabilistic models for retrieving fewer relevant documents. In: SIGIR 2006, pp. 429–436. ACM, New York (2006)
2. Eisenberg, M., Barry, C.: Order effects: A study of the possible influence of presentation order on user judgments of document relevance. JASIS 39(5), 293–300 (1988)

3. Feynman, R.P.: The concept of probability in quantum mechanics. In: Proc. 2nd Berkeley Symp. on Math. Statist. and Prob., pp. 533–541. Univ. of Calif. Press, Berkeley (1951)
4. Maron, M.E., Kuhns, J.L.: On relevance, probabilistic indexing and information retrieval. J. ACM 7(3), 216–244 (1960)
5. Robertson, S.E.: The probability ranking principle in IR, pp. 281–286. Morgan Kaufmann Publishers Inc., San Francisco (1997)
6. van Rijsbergen, C.J.: Information Retrieval, Butterworths (1975)
7. Zhai, C.X., Cohen, W.W., Lafferty, J.: Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In: SIGIR 2003, pp. 10–17. ACM, New York (2003)

Written Texts as Statistical Mechanical Problem

Kostadin Koroutchev[1,⋆], Elka Korutcheva[2,⋆⋆], and Jian Shen[1,⋆⋆⋆]

[1] EPS, Universidad Autónoma de Madrid, Spain
k.koroutchev@uam.es, jian.shen@estudiante.uam.es
[2] Depto. de Física Fundamental, UNED, Spain
elka@fisfun.uned.es

Abstract. In this article we present a model of human written text based on statistical mechanics consideration. The empirical derivation of the potential energy for the parts of the text and the calculation of the thermodynamic parameters of the system, show that the "specific heat" corresponds to the semantic classification of the words in the text, separating keywords, function words and common words. This can give advantages when the model is used in text searching mechanisms.

1 Introduction

The evolution of web search engines during the last decades made the statistical analysis of text an intensively developing area. In the information retrieval (IR) theory this analysis is applied with remarkable success [1]. The statistical consideration can be explicitly oriented toward the solution of document retrieval from large documents collections [2,3,4]. Another statistical consideration is centered on the Zipf law, but we do not consider this aspect of text statistics in the present article.

The model we purpose consists of a text and a vocabulary. The whole text is written in the same language. The vocabulary V is defined as all the words contained in some huge collection of texts, written in that language. The different kind of words we analyze in the text are the *specific terms* or *keywords*, that have higher occurrence in the text than in the common language and are essential terms to explain the meaning of a particular text. For example "IR" is a keyword in this article. The *function words*, which by themselves have little lexical meaning but are essential for expressing the language structure. Basically they express grammatical elements. A typical example of a function word in English are the words "the" and "and". Finally, the third category consists of the *common words*, which have the same frequency in wide range of texts in this language and represent the common lexical elements.

We show that these categories of words can be effectively separated using the thermodynamic parameters.

[⋆] Supported by Spanish MEC grant TIN 2007-66862-C02-01.

[⋆⋆] Supported by Spanish MEC grant DGI.M.CyT FIS2005-1729.

[⋆⋆⋆] Supported by "Beca de colaboración" MEC 2008.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 241–248, 2009.

2 The Model

Our model consists of a vocabulary of length L_v, a text of length L_t, and the words in the text $\{w\}$, which are also present in the vocabulary. The corresponding number of occurrences of the word w is $n_t(w)$ and $n_v(w)$ in the text and in the vocabulary, respectively. In order for the text and the vocabulary to have equal length, we introduce some standard text length L_0 and normalize the number of occurrence of w according to this length: $N_t(w) = L_0\frac{n_t(w)}{L_t}$, $N_v(w) = L_0\frac{n_v(w)}{L_v}$.

We introduce an order parameter $m(w)$ that corresponds to the matching between the occurrence of the word in the text and in the vocabulary. The thermodynamic approach consists now in defining the energy of the interaction $E(w) = E(m(w), N_t(w), N_v(w), L_0)$ between the vocabulary and the text, expressed in terms of the order parameter $m(w)$, which will be defined rigorously further. We are looking for energy that has its minimum if the frequency distribution of the word in the text and in the language coincide.

As a first approximation, in this paper, we assume that the words are independent, e.g. that there is no interaction between different words. As we will see further, even this approximation our method captures pretty well the function words and keywords.

According to our model, the probability $P(m)$ of the state with value of the order parameter equal to m is [6]: $P(m) \propto G(m, N_t)\exp(-\beta E(m, N_t, N_v, L_0))$, where $E(m, N_t, N_v, L_0)$ is the energy of that state. $G(m, N_t)$ is the number of degenerations (combinatorial factor) of the states and β is the inverse temperature $\beta \equiv 1/T$. The number of degenerations is just the number of ways we can select m words out of a set of N_t words, e.g. $G(m, N_t) = \binom{N_t}{m}$. Note that this number is strictly zero if $m > N_t$.

3 Frequency of a Single Word

Let us consider the frequency of occurrence x of a single word w in a text with length L. We suppose that L is large enough in order to have $x \gg 1$. The question is, what is the probability distribution of x ?

The usual hypothesis is that the distribution is binomial or a mixture of binomials that corresponds to some urn process [5]. More sophisticated models suppose that the distribution is a mixture of binomial (when the word is not used as a keyword) plus a flat distribution (when the word is used as a keyword) [7].

However, the answer of what the distribution of x is can be given only by empirical argument investigating a large repository of texts. We have found that the distribution is far from binomial. As an illustration, in Fig. 1(left) we give the frequency distribution of the word “the” in the Gutenberg collection [8] of texts, with $L = 10000$. This word is practically impossible to be used as a keyword and therefore one can assume that the distribution would be simply binomial. It is clear that the distribution is not binomial; it is highly skewed and far away from the binomial distribution with that frequency [9]. Moreover, in the case of the word ”the”, when $L = 10000$, the distribution has a mean $n = 628$ and the

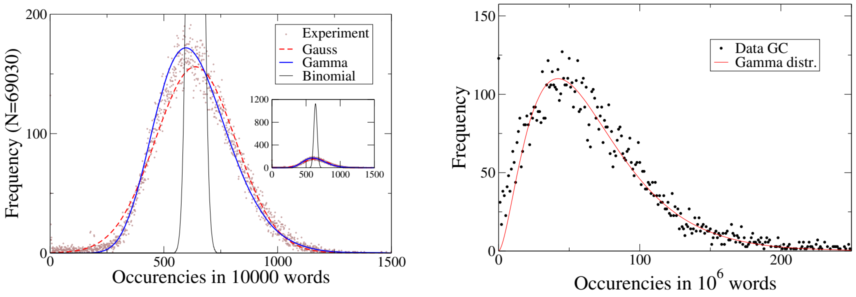


Fig. 1. (Color online) Left: Frequency distribution of the word "the" in 10 000 consecutive words of the corpus with different fits. Right: Frequency distribution of "house" in 100 000 consecutive words of the corpus with Gamma fit.

standard deviation is $\sigma = 128$, so $\sigma^2/n \gg 1$ and the distribution cannot even theoretically be a binomial one. The same is also true in the case of other words, for example "house" Fig. 1(right) with $L = 100000$.

We have also done an extensive analysis by using the British National Corpus [10], a set of about 19000 English texts chosen from the Gutenberg collection and Chinese corpus with some 10^8 words. We divided the text into segments with sufficient length L in order to have $L\bar{x} \gg 1$, where we denote with $\bar{x}$ the mean of x. By analyzing these text repositories we have found that the distribution of x is close to the gamma distribution:

$$P(x; a, b) = x^{a-1} b^a e^{-bx} / \Gamma(a), \tag{1}$$

where a and b are the parameters of the distribution. In all cases where $\bar{x} \gg 1$ we have found that the gamma distribution describes the data better $(P > 0.6)$ than the binomial one $(P < 0.3)$.

In order to build a thermodynamic theory we need the asymptotic behavior of the Gamma distribution on a large text database. To achieve this we replicate the text s times and consider the limit $\lim_{s\to\infty}[\log P(sx; w; sa, b)]/s = a - bx - a\log a + a\log x + a\log b$. Using that the mean of x is $\bar{x} = a/b$, we obtain the following expression $E_p(x; w) = -\log P(x) = b\bar{x}\left[\frac{x}{\bar{x}} - 1 - \log\left(\frac{x}{\bar{x}}\right)\right]$ for the asymptotic behavior of $\log P(x)$. Here E_p can be regarded as a potential energy of the word w in the language. The logarithmic term corresponds to the entopic part of the energy, while the linear one accounts for the excess of words of a given type in the text.

4 The Free Energy

In the following we will use the statistical mechanics approach by defining the corresponding partition function for the problem. From general considerations, the partition function is given by the following expression $Z = \sum_s \exp(-\beta E_s)$, where by s we label the states that the system occupies and E_s is the energy of the system corresponding to these states.

In our case, the partition function is:

$$Z(w, \beta) = \sum_{m=1}^{N_t} \exp(-\beta E_{tot}(m, N_t)). \tag{2}$$

Here

$$E_{tot}(m, N_t) = -\frac{1}{\beta} \log \binom{N_t}{m} + N_v b \left[\frac{m}{N_v} - 1 - \log \left(\frac{m}{N_v} \right) \right] \tag{3}$$

is the total energy corresponding to some word w and we have included the degeneration factor inside the exponent. The full energy of the text is a sum over all the words of the text.

The equation for the order parameter m can be obtained by using the saddle-point method $\frac{dF}{dm} = 0$, [11] that gives us the final expression for the order parameter m: $m = N_t \frac{b\beta N_v/N_t}{b\beta N_v/N_t + W(b\beta N_v/N_t \ e^{b\beta - b\beta N_v/N_t})}$, where $W(.)$ is the Lambert W-function.

Further, we consider the rest of the thermodynamic quantities: the entropy S for a single word, $S \equiv -\frac{\partial F}{\partial T}$ and the "specific heat" $C_V = -T \left(\frac{\partial^2 F}{\partial T^2} \right)_V$.

In the context of the statistical model of texts, the last quantity can be interpreted in the following way: if C_V is high for a given word, then replacing this word by another one, or omitting it, will introduce a relatively large distortion in the text meaning, leading to a significant change of the total energy. On the other hand, replacing a word with negligible C_V will have no significant consequence on the text.

5 Numerical Experiments

To check the above results experimentally on real texts, we used several corpora of texts. First, we used the British National Corpus(BNC), as a standard and equilibrated corpus of English texts with some 10^8 words. Second, we used a collection of about 19000 English texts of the Gutenberg collection (GC) with size 5.10^7 words. To check specific domains we used single articles, as well as a collection of 500 articles from the non-linear physics archive (NL) offered by the arXiv repository http://arxiv.org.

Fig. 2 shows a typical behavior of C_V for keywords, for function words and for common words. As the function words have much higher frequency of occurrence, one can expect that they will have a predominant role in the specific heat. However, this is not observed. The specific heat of the keywords **in a text** is much higher than that corresponding to the function words. Even smaller specific heat is carried by the common words. These results can be interpreted as an indication that the most vulnerable speech parts are the common words, and the most resistant ones are the keywords.

Considering all the words with their respective parameters $\bar{x}$ and b, we can numerically calculate the free energy F, the entropy S and the specific heat C_V for the whole text. The result for C_V is shown in Fig. 3.

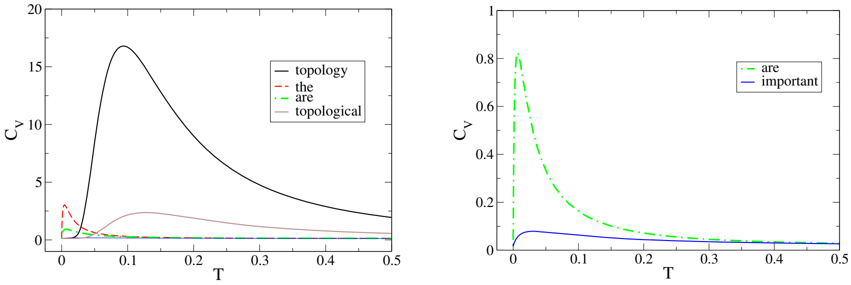


Fig. 2. C_V for different words of one and the same text. The upper two curves of the left panel represent two keywords of a given text ("topology" and "topological"). The lower curves of the left panel represent two functional words ("the" and "are"). On the right panel the curve of "are" is zoomed in order to represent also the typical common word "important".

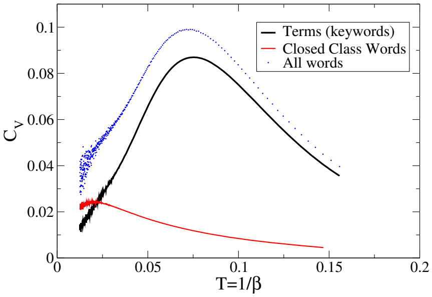


Fig. 3. (Color online) Experimentally measured C_V on a single text. The part of the C_V corresponding to the common words is very small to be shown in this scale.

What is observed experimentally (Fig. 3) is the lack of a well-pronounces maxima of C_V for the function words, less expressed maxima for the common words and well pronounced maxima for the keywords. The function words express the structure of the language, e.g. represent its grammatical structure. The keywords, on the other hand, are expressions of the semantic and the pragmatic structure of the text. The typical error in C_V measurement is less than 0.01 and we can see from the figure that we can separate the three classes of words at $T = 0.07$ using C_V.

6 Comparison with TREC Results

In order to compare our model with other models from the Information Retrieval area, we use the most popular IR collection for the purpose — the text retrieval conference (TREC) texts collection [12] in its tenth edition (WT10g collection). The goal of the TREC contest consists in finding out the texts relevant to answer some query. There are several hundred thousand texts with very different lengths and styles. We use the title, usually of less than 10 words, as a query.

We would like to note that our model in its present form is adapted well to large texts, with length of several thousands of words and not to the texts of the TREC collection, where documents with length 10 to 100 words are not an exception.

Following the standard IR approach [2], we introduce a query and score the results for each word of the query. For this aim, we need first a document collections specific term, which characterizes the document collection as a whole, and second, a document specific term, which characterizes the document, where the word is present. The first term can be extracted only having in mind the specific collection of texts. This document collection specific term is normally accepted to be the Inverse Document Frequency (IDF) [3,4] term:

$$IDF(w_i) = \log \frac{M - n(w_i) + 0.5}{n(w_i) + 0.5}, \tag{4}$$

where M is the number of documents in the collection and $n(w_i)$ is the number of documents, where the word w_i is present.

In our model we compare the text with the language as a whole, considering the corpus as representative for the language and do not distinguish between the individual documents in it. Therefore we need some approximation of the term above. Assuming Poisson distribution of the texts' lengths in the collection with a mean length λ and uniform words occurrence in the whole collection, we approximate the IDF term with its expectation value $F_1(w_i)$:

$$F_1(w_i) \equiv \log(\lambda e/N_v). \tag{5}$$

Here N_v is supposed to be normalized to one word ($L_0 = 1$).

One of the best known and best-scoring functions employed in the IR is the so called BM25[2,3] score with a document specific factor[1] equal to: $(K_1 + 1)N_t/(N_t + K_1 b + K_1(1 - b)L_t/\lambda)$. In this expression, the length of the text L_t is explicitly accounted. In the present version of our model, we effectively ignore any document-length effect, rescaling the document to the size of the vocabulary. Therefore, the fair comparison should suppose that all documents have one and the same length, fixing L_t to the value of λ. Effectively this gives BM15 [4].

Summarizing, the score formulas we use for the comparison are IR based:

$$S_{BM25} = \sum_i IDF(w_i) \frac{2.25 N_t(w_i)}{N_t(w_i) + 0.5 + 0.75 L_t/\lambda}, \tag{6}$$

$$S_{BM15} = \sum_i IDF(w_i) \frac{2.25 N_t(w_i)}{N_t(w_i) + 1.25},$$

and thermodynamically based:

$$S_m = \sum_i F_1(w_i)(1 - m(w_i)/N_t(w_i)) \tag{7}$$

[1] We suppose that we have no a-priori information about the query. The parameters we use are: $b = 0.6$, $K_1 = 1.25$.

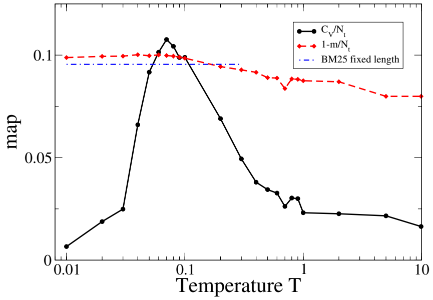


Fig. 4. (Color online) Comparison of MAP for S_m, S_{C_V} and S_{BM15} for different temperatures T

Table 1. Comparison between the different score functions: S_m, S_{C_V} and the mix between them. The score function S_{BM25} is the BM25 criteria (see the text) and S_{BM15} is BM25 without accounting for the length of the text. Our method gives slightly better results than BM15.

Criterion	MAP	ircl(0)	ircl(0.1)	P@5	P@10	P@100
S_{BM25}	0.1446	0.5456	0.3670	0.336	0.280	0.134
S_{BM15}	0.0955	0.3884	0.2261	0.192	0.172	0.105
S_m	0.1002	0.3959	0.2318	0.184	0.170	0.100
S_{C_V}	0.1076	0.3338	0.2203	0.176	0.158	0.095
S_m and S_{C_V}	0.1094	0.4092	0.2397	0.188	0.186	0.105

and

$$S_{C_V} = \sum_i F_1(w_i)(C_V(w_i)/N_t(w_i)), \qquad (8)$$

where the words w_i are the words of the query. The score function S_m uses document specific term $1-m/N_t$ and finally the score function S_{C_V} uses C_V/Nt. As performance measure we use the criteria adopted by the TREC Conference [12].

In Fig.4 we represent the Mean Average Precision (MAP) criteria for the three cases: $1-m/Nv$, C_V and BM15 as a function of the temperature. The peak value of the MAP occurs at the same temperature as the maxima for the keywords.

The comparison of the methods is given in Table 1. The comparison classifies it marginally better compared to BM15. From the table, one can see that the thermodynamic approach is rather good for the analysis of texts, compared to the best of the IR methods.

7 Conclusion

In the present article we propose a statistical physics approach for the analysis of human written text. By introducing the concept of energy of interaction between the text and the corpus (the language), and taking into consideration a realistic distribution of the words inside a given large text corpus, we derive the thermodynamic parameters that describe the system.

The behavior of the specific heat of the system is different for the different kinds of words (keywords, function words and common words). It is universal and independent for the selected text and can be used for tasks when we ought to separate different kinds of classes of words.

We compared the thermodinamical model with one of the best IR approaches that use the same "bag of words" approximation for the text. Regarding the IR performance our results are very competitive.

In our opinion the thermodynamic consideration can have an advantage because it is based on energy, which is additive quantity and it is relatively easy to amplify the model with interactions between the text's parts corresponding to the grammar and the semantic.

References

1. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley, Essex (1999)
2. Amati, G., van Rijsbergen, C.J.: Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness. ACM Trans. Inf. Syst. 20, 289–357 (2002)
3. Robertson, S., Sparck Jones, K.: Relevance Weighting of Search Terms. J. Am. Soc. Inf. Sci. 27, 129–146 (1976)
4. Robertson, S.: Understanding Inverse Document Frequency: On theoretical arguments for IDF. Journal of Documentation 60(5), 503–520 (2004)
5. Baayen, R.H.: Word Frequency Distributions. Kluwer, Dordrecht (2001)
6. Beck, C., Schloege, F.: Thermodynamics of Chaotic Systems. Cambridge University Press, Cambridge (1993)
7. Labbé, C., Labbé, D., Hubert, P.: Automatic Segmentation of Texts and Corpora. Journal of Quantitative Linguistics 11(3), 193–216 (2004)
8. The Gutenberg collection, http://www.gutenberg.org
9. Shen, J., Koroutchev, K.: Message Exchange and Energy Model of Text. Technical report UAM Spain (June 2008)
10. The British National Corpus, Version II, Distributed by Oxford University Computing Service on behalf of the BNC Consortium (2001), http://www.natcorp.ox.ac.uk
11. Erdelyi, A.: Asymptotic Expansions. Dover, New York (1956)
12. Voorhees, E., Harman, D. (eds.): TREC: experiment and evaluation in information retrieval. MIT Press, Cambridge (2005)

What Happened to Content-Based Information Filtering?

Nikolaos Nanas[1], Anne De Roeck[2], and Manolis Vavalis[1]

[1] Laboratory for Information Systems and Services
Centre for Research and Technology - Thessaly (CERETETH)
{n.nanas,m.vavalis}@cereteth.gr
[2] Computing Department, The Open University
a.deroeck@open.ac.uk

Abstract. Personalisation can have a significant impact on the way information is disseminated on the web today. Information Filtering can be a significant ingredient towards a personalised web. Collaborative Filtering is already being applied successfully for generating personalised recommendations of music tracks, books, movies and more. The same is not true for Content-Based Filtering. In this paper, we identify some possible reasons for the notable absence of a broad range of personalised information delivery and dissemination services on the web today. We advocate that a more holistic approach to user profiling is required and we discuss the series of still open, challenging research issues raised.

1 Introduction

The World Wide Web is becoming a network of transmitters and receivers of information. With the advent of technologies as simple as Really Simple Syndication (RSS), anyone can broadcast ideas, thoughts, comments, video clips and more, potentially to millions of individuals. In the context of Web 2.0, with social networking and the general culture of participation, the horizontal, peer-to-peer dissemination and exchange of information, brings about a radical alternative to traditional broadcasting models, which can unleash far reaching social change. But when there are millions of information transmitters/receivers, issues of sustainability arise. It is just impossible to keep up with the gigabytes of information that can be delivered to one's PC, mobile phone, or other networked device, or to guard effectively against spam, or unwanted communication. On the other hand, individual publishers have no way to ensure that an idea or opinion, once broadcasted, will reach the right audience. Personalised information filtering could alleviate this dual problem.

Information Filtering (IF) is a mature research domain. A distinction is usually made between Content-Based Filtering (CBF) and Collaborative Filtering (CF). Both types of IF aim for personalised information delivery, but do not share the same level of success. CF has been successfully applied to generate personalised radio stations (Last.fm) and to recommend books (Amazon.com), or movies (Netflix.com). In contrast, we are not aware of any broadly adopted,

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 249–256, 2009.

personalised information delivery service based on CBF that is publicly available on the Web today. This absence is puzzling.

So what happened to CBF? In this paper we attempt to elucidate this interesting research issue. We start with a clear definition of personalised IF, in the context of current web trends and technologies, and we specify its requirements. These provide a framework for a qualitative analysis of the approaches that shaped research in CBF, rather than a comprehensive review of existing models and algorithms. We advocate that personalised IF is a complex, dynamic and user-dependent problem, with its own particular characteristics and requirements, that still remains unresolved. It opens new research challenges and attracts novel approaches. Successful personalisation of information dissemination could have a fundamental impact on the Web and beyond. It is a research direction worth pursuing.

2 The Problem and Its Requirements

Although the problem of information overload has intensified through web developments such as Usenet newsgroups, Forums, Mailing Lists and RSS feeds, research interest in CBF has declined recently[1]. Social Networking and the Web 2.0 culture of participation, are causing an explosion in user generated content, which is no longer just textual. Audiovisual content is increasing rapidly. In parallel, with popular annotation techniques, such as tagging, the amount of metadata has also grown. In this landscape, personalisation could enhance significantly the horizonal dissemination of information taking place on the web today.

CBF, and IF in general, can play an important role in personalising the web, but this requires a new holistic approach towards user profiling that takes into account the current state of the web: a user profile should no longer be a filter blocking unwanted information, but be seen as a personalised interface between any individual and the web. In general, a user profile can be defined as a computational model that can continuously and effectively evaluate the relevance of any information item, or source, to the interests/needs/context of a particular user (or group). This implies personalised IF that is media-independent, multimodal, scalable, dynamic and viable. We argue that existing approaches to IF cannot easily cope with the above requirements of user profiling, mainly due to a restricted view of IF as a specialisation of other domains, which has lead to simplifying assumptions. The problem has been adapted to the existing solutions rather than the other way around. As a result many challenging research issues have remained unaddressed in IF.

3 The Profile and the Query

Since the seminal paper “Information Filtering and Information Retrieval: Two Sides of the Same Coin?” by Belkin and Croft [3], it has become a common

[1] This is highlighted by the removal of the TREC Filtering track since 2001.

belief that "there is little difference between the two [IR and IF] at an abstract level" and that "most of the issues which appear at first to be unique to information filtering, are really specialisations of IR problems". This belief made the Vector Space Model a popular choice for IF, representing both the documents and the profile as (weighted) keyword vectors on a common n-dimensional space (e.g., [1,15]). The profile was treated as a persistent query, representing "information needs which are relatively stable over relatively long periods of time", that the user had to construct and subsequently refine [3].

However, unlike a query, the user profile neither has to be, nor should it be user specified. A query is a temporary representation "destroyed" at the end of an information seeking episode, whereas, a user profile is a long term construct that can be generated, maintained and improved automatically. This is a significant distinguishing characteristic with important consequences. The user profile should not contain just words, but a multitude of features extracted from information items. These may include "silent" features that a user would not think of adding to a query. Profile features are not user specified and do not need to make sense to the user. They can be any comparable entity that can be extracted from and matched against information items. Lower level features extracted from a variety of media, can be as informative as higher level ones. Unlike a user generated query, an automatically constructed profile may incorporate all the necessary features for evaluating the relevance of music tracks, video clips and more. Furthermore, a user profile can store more information about the users interests/needs/context than a query can carry and thus can be more specific. For instance, it can take into account statistical dependencies between terms in text, but research in IF tends to ignore them [10].

There are other problems with aligning profiles with queries. A query is a short description of a specific information need, but a users long term interests cannot be similarly focused. A user may be interested in a diverse set of subject areas (or topics). A (weighted) keyword vector cannot effectively represent more than a single topic of interest, because it ignores the context of words (or of features in general): any possible combination of words in a keyword vector is equally represented[2]. To capture the whole range of user interests, a profiles keyword vector would have to incorporate a large number of words (features). However, in this model, the number of possible word combinations increases exponentially with each addition and the profile becomes ambiguous. This may explain why research in IF tends to break up the problem into a separate keyword vector for each topic of interest to the user (see for instance [1,15]). The approach assumes that topics of interest can be easily identified and distinguished, but as we will further discuss in section 5, this is an unrealistic simplification.

Importantly, unlike a query, a user profile does not *retrieve* information items, but evaluates them, so in IF the bulk of computation can be moved from the index to the profile. Documents do not have to be represented as a "bag of words": the user profile can be given access to each actual information item,

[2] Note also, that although indexing can take the order of words in text into account, a weight keyword vector cannot.

its content and the associated metadata. A user profile can assign one, or more scores to an information item, or even to portions of it, which can be used for ranking along various (user sensitive) relevance dimensions, related to content, context, community impact, timeliness etc., and also, to guide the user to the most interesting parts of an information item, or to highlight the most relevant hyperlinks.

Overall, a user profile can be a computational entity with a much broader scope than a query. Such a holistic profile could incorporate a large number of features of different types, represent the complete range of evolving user interests, perform a variety of evaluation functions on information items of any media, and support a series of personalisation services. So far, this broad view of IF has been missing.

4 The Profile, the Classifier and the Recommender

Some work treats IF as a specialisation of Text Classification (TC). A separate classifier is built for each topic of interest to the user and is used to assess the relevance of documents, or to calculate the probability that a document belongs to a specific topic [7]. Non-linear Neural Networks have been deployed for building multi-topic classifiers [14], but they assume that the topics of interest are predefined and fixed. Again, this assumption is counterintuitive for IF because the users interests are fluid, multi-faceted, interwoven and not easily mapped to concrete subject areas. So, unlike multi-class categorisation, it is not necessary, perhaps even impractical, for an information filter to be able to pin down the topic (or topics) that information items belong to. For IF, their essential property is their relevance to the users interests (and/or context).

TC typically works from a predefined set of subject categories (topics), each associated with a large collection of documents preclassified by human experts. Machine learning algorithms then train a classifier for each topic category from its document set. Once built, a classifier specialises to its topic and usually remains unaltered over time. In contrast, in IF it is safer to assume that there are no initial feedback (training) documents. The IF system should be able to start from an empty profile that learns continuously thereafter, from the interaction between user and system. Hence, machine learning algorithms, such as Support Vector Machines (SVMs) [5] that require a large training collection, or lack an inherent online mode of operation, are probably unsuited to the task.

CF works from a matrix of user ratings assigned to information item (e.g., movies, music tracks, or books). CF suffers from some known problems, especially when this user-item matrix is sparse and correlations between users (or between items) cannot be estimated with confidence [2]. The “ramp up” problem refers to the difficulty in making recommendations to a user who has rated a small number of items, or to the difficulty in recommending an item which has not yet received enough ratings [6]. Consequently, CF cannot scale up easily to dynamic domains like news publishing, involving many and regularly updated information items. In this latter case, the dominant current approach is to identify popular news stories on the web by voting[3], rather than issueing recommendations.

Researchers in CF have attempted to alleviate these problems through a hybridisation of CF and CBF, which does not require rated information items, or a community of users with overlapping interests. Hybrid IF uses both content-based and collaborative profiles, combining them appropriately when estimating recommendations, or exploiting content-based features when calculating correlations between users (or between items) [4]. One alternative approach to hybrid IF still remains unexplored. Hybridisation could be achieved by incorporating social features in a content-based profile. Where a feedback document has received rating from users in a community, these could be incorporated in the profile, in the same way as textual features. This natural hybridisation of content-based and collaborative filtering overcomes the earlier problems, and combines the strengths of each individual approach. Such a hybrid profile could adaptively evaluate and recommend information items to a user, based on both content and social features. To our knowledge, adaptation to interest changes has been generally ignored in CF research.

5 The Complexity and the Dynamics

The most challenging aspect of IF is its dynamic nature. Both the users interests and the topicality of received information, change over time. The user profile must track the user's interests over time, to maintain a satisfactory level of performance. Dissatisfied users will abandon the system and further profile adaptation will cease. This means that in IF, unlike IR and TC, the systems ability to maintain performance over time is probably as important as filtering accuracy. In other words, we should be less concerned about improving filtering accuracy by some small percentage and more about maintaining a satisfactory performance level indefinitely. This is a complex and dynamic problem that has not been tackled comprehensively so far[4].

Learning Algorithms (such as Rocchio Algorithm, or variations [11]) have proved a popular choice for tackling profile adaptation, typically by improving a profile representing a single, predefined topic, over time. Learning coefficients define the effect that each feedback document (positive or negative) has on the profile. In Reinforcement Learning, the coefficient values are reduced as the number of processed feedback documents increases, so that what has already been learned about the topic of interest is maintained [12]. Other work combined short-term and long term profiles and used different learning parameters for each of them [15]. In all cases, system parameters define how the profile adapts and may have to be fine tuned. For example in [11], the authors run multiple instances

[3] See for instance digg.com or reddit.com

[4] The use of a research terminology referring to topics should probably be avoided: it is a very subtle concept in the case of IF, and the key point is whether the user profile is able to capture continuous shifts, from (implicit or explicit) feedback, within the stream(s) of information items that the system is monitoring.

of Rocchios algorithm in parallel, and choose the best values for the coefficients in an online fashion. It has been argued [13], that learning algorithms cannot deal effectively with radical interest changes. This is probably justified because these algorithms have been devised with the problem of optimising a single-topic representation in mind and they typically lack an inherent mechanism for introducing a new area of interest in the profile, or for removing a waning one.

In contrast, biologically inspired solutions propose an inherently dynamic solution to profile adaptation. Organisms constantly have to adapt to, and learn from, a dynamically changing environment. Genetic Algorithms (GAs) have been applied for adapting a population of profiles to interest changes. According to [8] however, GAs suffer when applied to IF, because they tend to converge on a single optimum (topic of interest), gradually loosing population diversity. The emerging field of Artificial Immune Systems proposes another biologically-inspired solution. The immune system needs to distinguish between cells that belong to the host organism and external bacteria, or viruses and offers a computational metaphor for building user profiles that can distinguish between relevant and non-relevant information. This analogy is already being explored with some promising first results[5]. In general however, biologically inspired approaches have not received much attention from researchers in IF. It is characteristic that, to our knowledge, none of the participants of TRECs adaptive filtering track have adopted such an approach. Fully addressing profile adaptation in all its complexity lies on the critical path towards successful personalised IF. Much more effort should be put in dealing with the complex and dynamic nature of the problem.

6 The User and the Profile

Maintaining the users interest in the system is critical to viable IF, and interfaces have a role to play. There is no body of work around this aspect of IF (unlike IR or browsing). For instance, is a single ranked list of information items an appropriate way to present filtering results? On the one hand, a ranked list leaves it to the user to decide when to stop looking down the list. Furthermore, to deal with RSS feeds, a user profile has to cope with batches of the most recently published documents in each feed. On the other hand, how can a single list highlight items on multiple topics of interest and what happens when information items about topics have different publishing rates? Interface design should allow a user to manage and monitor multiple information sources, publish to multiple recipients and trigger/capture implicit and explicit feedback. Interfaces may counteract the tendency of some IF approaches to overspecialise on topics the user has already viewed, by encouraging feedback to external information items not presented by the IF system itself. In this way, a user can guide the profile towards new areas of interest. Finally, there is the question whether IF is suitably treated as a stand-alone application, or whether there is merit in integrating personalised IF with other existing web applications for information dissemination and exchange, with the user profiles running in the background.

[5] For a comprehensive review of Evolutionary and Immune Inspired IF see [9].

7 The Real and the Virtual

The most common approach to the evaluation of IF systems is to perform simulated experiments, based on a document collection, which has been pre-classified according to a number of topic categories. The well established TREC[6] has standardised methodologies for performing such simulated experiments. TREC's Adaptive IF track evaluates the ability of a user profile representing a single topic to adapt to variations, over time, in the content of documents that have been assigned to this topic. It does not simulate the variety of changes that can occur to a user's interests. There are still no standard evaluation methodologies that encompass the full complexity and dynamics of profile adaptation, in a fair and undisputed way. Simulated experiments should test the ability of profiles to maintain an accurate representation of the user's multiple interests over time, and do so effectively, but also efficiently. If an IF system is not fast enough then it is bound to dissatisfy the user. Simulated experiments are an important research tool, but the ultimate criterion of success for a personalised IF system is its adoption by real web users. To tackle the problem of personalised IF comprehensively, we should start testing our systems in real world situations.

8 The Future of IF

Web personalisation will have a radical impact in the way information is disseminated and consumed. We believe that IF and user profiling in particular, has an important role to play in personalising the web. However, unlike CF, which has been successfully applied for personalised recommendations, CBF has not produced similar success stories. In this paper, we have tried to identify some of the reasons for this lack of development. We conclude that there is still much room for innovation in IF, but it requires a fresh attitude towards user profiling and a reformulation of the research agenda associated with it. In summary, research in IF should concentrate on:

- developing user profiling models that are media independent and can incorporate any features, or metadata, that can be extracted from, or have been assigned to, information items.
- developing dimensionality-resistant user profiling models that can incorporate a large number of features, but can also scale up to a large number of individual users.
- developing algorithms for adapting user profiles to the continuous changes in user interests, so that profiles can maintain their viability.
- expanding the scope of user profiling to a variety of personalised information dissemination services.
- designing appropriate user interfaces for personalised web systems.
- designing appropriate experimental methodologies that reflet the complexity and dynamics of the problem.
- developing working prototypes that are used and evaluated in real situations.

[6] http://trec.nist.gov/

Web personalisation will change the media landscape, have a fundamental societal impact, and enhance collaboration, creativity and collective actions. It has the potential to boost collective intelligence. This is a prospect worth pursuing and the science of IF has still a lot to offer towards this end.

References

1. Amati, G., D' Aloisi, D., Giannini, V., Ubaldini, F.: A framework for filtering news and managing distributed data. Journal of Universal Computer Science 3(8), 1007–1021 (1997)
2. Balabanovic, M., Shoham, Y.: Combining content-based and collaborative recommendation. Communications of the ACM 40, 66–72 (1997)
3. Belkin, N.J., Croft, W.B.: Information filtering and information retrieval: Two sides of the same coin? Communications of the ACM 35(12), 29–38 (1992)
4. Burke, R.: Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction 12(4), 331–370 (2002)
5. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998)
6. Konstan, J.A., Riedl, J., Borchers, A., Herlocker, J.L.: Recommender systems: A grouplens perspective. In: Recommender Systems. Papers from 1998 Workshop. Technical Report WS-98-08, pp. 60–64. AAAI Press, Menlo Park (1998)
7. Mladenic, D.: Using text learning to help web browsing. In: 9th International Conference on Human-Computer Interaction (HCI International 2001), New Orleans, LA, pp. 893–897 (2001)
8. Nanas, N., De Roeck, A.: Multimodal dynamic optimisation: from evolutionary algorithms to artificial immune systems. In: de Castro, L.N., Von Zuben, F.J., Knidel, H. (eds.) ICARIS 2007. LNCS, vol. 4628, pp. 13–24. Springer, Heidelberg (2007)
9. Nanas, N., De Roeck, A.: A review of evolutionary and immune inspired information filtering. Natural Computing (2007)
10. Nanas, N., Vavalis, M.: A "bag" or a "window" of words for information filtering. In: Darzentas, J., Vouros, G.A., Vosinakis, S., Arnellos, A. (eds.) SETN 2008. LNCS (LNAI), vol. 5138, pp. 182–193. Springer, Heidelberg (2008)
11. Pon, R.K., Cárdenas, A.F., Buttler, D.J.: Online selection of parameters in the rocchio algorithm for identifying interesting news articles. In: WIDM 2008: Proceeding of the 10th ACM workshop on Web information and data management, pp. 141–148. ACM, New York (2008)
12. Seo, Y., Zhang, B.: A reinforcement learning agent for personalized information filtering. In: Intelligent User Interfaces, New Orleans, LA, pp. 248–251 (2000)
13. Webb, G.I., Pazzani, M.J., Billsus, D.: Machine learning for user modeling. User Modeling and User-Adapted Interaction 11, 19–29 (2001)
14. Wermter, S.: Neural networks agents for learning semantic text classification. Information Retrieval 3, 87–103 (2000)
15. Widyantoro, D.H., Ioerger, T.R., Yen, J.: An adaptive algorithm for learning changes in user interests. In: ACM/CIKM 1999 Conference on Information and Knowledge Management, Kansas City, MO, pp. 405–412 (1999)

Prior Information and the Determination of Event Spaces in Probabilistic Information Retrieval Models

Corrado Boscarino and Arjen P. de Vries

Centrum Wiskunde & Informatica (CWI), Science Park 123,
1098 XG Amsterdam, The Netherlands
corrado@cwi.nl, arjen@acm.org

Abstract. A mismatch between different event spaces has been used to argue against rank equivalence of classic probabilistic models of information retrieval and language models. We question the effectiveness of this strategy and we argue that a convincing solution should be sought in a correct procedure to design adequate priors for probabilistic reasoning. Acknowledging our solution of the event space issue invites to rethink the relation between probabilistic models, statistics and logic in the context of IR.

1 Introduction

Information Retrieval (IR) can be distinguished from other Information Access (IA) classes of techniques, like that to which deterministic database access belongs, by being mainly concerned with uncertain knowledge, at least when assuming a notion of relevance of documents with respect to the subjective and unpredictable opinion of that particular human agent that is supposed to have issued the query. Acknowledging the presence of uncertainty leads naturally to probabilistic models as the chief mathematical description of uncertain information about reality. Probability theory is the framework within which we can make precise statements about imprecise features of the world, or, in slightly different terms, about features that admit multiple precisifications. As long as the human user is considered to provide the ultimate metric of success in information retrieval, alternative approaches differ mainly in how they represent uncertainty. These representations do not stand apart because they question the validity of probability theory, but in their usage of the theory. In a formally correct model there will be one or more stochastic variables representing the uncertain link between the output of the retrieval system and the end user's satisfaction. However, given the knowledge of the parameters and of the structure of two or more models that differ in non trivial features of their design or in important presuppositions, comparing the alternative representations can be a very difficult task.

Lafferty and Zhai [1, pp. 1-10] introduce a probabilistic framework that allows to compare both the ranking and the assumptions of two well known retrieval models, the RSJ (Robertson-Sparck Jones) [4] model and the more recent language models [3]. They show that, at least in this particular case, the probabilistic semantics, that is the general rules of modeling uncertainty, are acknowledged

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 257–264, 2009.

in both approaches. They differ, however, in how they apply those rules: they factorise the same probabilities in two different ways and they may also possibly estimate the model's components with varying techniques.

Robertson [6] warns against the risk of calculating *rank equivalence*, or any other relationship between two or more probabilistic models, without considering the event spaces, that is the set of the real world objects, which we suppose our probabilistic model is about and upon which probabilities ought to be calculated. The mistaken assessment of event spaces can easily generate paradoxes and theory seems to support two mutually exclusive models of reality. Luk [5] acknowledges that, not the rules of probability itself are to blame, but the accuracy of their application to a particular problem, giving more emphasis to another link to the world, which the probabilistic model claims to represent: the estimation of the probabilities by means of a set of statistical components. Luk seems also to endorse a hierarchy of a statistical components model that provides the empirical content to an upper-level probabilistic model. While in Robertson the model's interface is lumped in the event space, Luk expands it into a separate model.

In this paper we argue that the paradox, the existence of which both Robertson and Luk agree on, albeit they propose different ways to circumvent it, is equivalent to the same marginalisation paradox that is known to the general public as the Monty Hall paradox [2]. The accepted solution to this paradox, however, cannot be expressed at the level of the probabilistic model considered by Robertson, nor by additionally considering the computational level as Luk does, but only by adding a logical level on top of the probabilistic one. Acknowledging our solution of the event space issue invites then to rethink the relation between probabilistic models, statistics and logic in the context of IR.

2 Event Spaces and Probabilistic Models

Arguing against the rank equivalence put forward by Lafferty and Zhai in [1, pp. 1-10], Robertson sets out to explain why a mixture of different event spaces could be problematic by introducing an analogous setup that is easier to understand. While an event space is often loosely defined as the set of the possible results of an experiment, the analysis brought forth by Robertson in [6] shows a more precise and at the same time more profound significance of this notion, albeit stated rather implicitly. This alternative definition of event spaces arises from an attempt to determine the conditions of applicability of the marginalisation equation, which allows the distribution of a variable Y to be calculated from the knowledge of the distribution of another variable X and the conditional probability of Y given X, as $P(Y) = \sum_X P(X)P(Y|_X)$. It is just because event spaces are linked to this fundamental relation of probability theory that this concept receives a more rigorous specification. At the same time we will see that constraining the physical objects, which the probabilistic model is about, to the domain of the marginalisation equation amounts to the reduction of the power of probabilistic reasoning; this is the limitation that Luk in [5] tries to remove.

For both authors, however, the event space becomes an ontological notion as the interface between an abstract description of the world and its extension.

2.1 Robertson's Argument

The first claim in [6] is that even models for simple situations, where estimation seems to become a straightforward process, can generate paradoxical cases in which the marginalisation equation does not yield the expected results. One example is about a very simple universe made of only two stars and three planets, where some of the stars s have a magnetic field ($x_{s_1} = 1$, $x_{s_2} = 0$) and some of the planets t that orbit around the stars also do ($y_{t_{11}} = 1$, $y_{t_{12}} = y_{t_{21}} = 0$): the problem is to calculate the probability $P(Y = 1)$ to find a planet with a magnetic field. In this situation we presume to be able to calculate the probability $P(Y)$ to find a planet which has a magnetic field by marginalisation on the probabilities $P(X)$ of a star to have a magnetic field, provided that we know the marginal probabilities $P(Y|_X)$. However, when we compare the result obtained by marginalisation with that calculated by simply counting the occurrences of a planet with a magnetic field and dividing by the total number of planets, we obtain two different results.

This mismatch, so goes Robertson's argument, is clearly paradoxical. The cause relies on a lack of expressive power in the notation used to express the marginalisation equation: we are unable to specify the objects in the real world to which the different probabilities apply. The distribution $P(X)$ applies to the event space $\mathcal{S}$ of stars, the conditional distribution $P(Y|_X)$ applies to the event space $\mathcal{T}$ of planets, while we need information about the full event space given by the cross-product $\mathcal{ST}$.

Robertson applies this result to the claim by Lafferty and Zhai [1, pp. 1-10] that classic probabilistic approaches like the RSJ model in [4] are equivalent at the level of the probabilistic model to the language modeling approach, although they may differ at the lower level of their statistical components, that is in how the probabilities are estimated. According to Lafferty and Zhai these two approaches correspond to two different factorisations in the marginalisation equation; the RSJ model results from the factorisation $P(D, Q|_R) = P(D|_{Q,R})P(Q|_R)$ and the language modeling approach from the factorisation $P(D, Q|_R) = P(Q|_{D,R})P(D|_R)$, where Q, D and R are the stochastic variables associated to the set of queries, to that of documents and to relevance, respectively.

Now consider that queries and documents play the role of stars and planets in the metaphor. Relevance is mapped onto both variables X and Y in the following way: relevance corresponds to the magnetic field and the division of labour between the two variables is related to the two approaches to IR that have been reviewed in [1, pp. 1-10]. The case of considering relevant queries that generate documents, that is the RSJ model, corresponds to stars with a magnetic field and it is accounted for by the X variable; the same applies for the language model, the planets and the Y variable.

In order for two probabilistic models to avoid the marginalisation paradox, the event spaces upon which the probabilities ought to be calculated should be the same; adapting the stars and planets example to the particular case of IR leads to the conclusion that the correct event space should be the most general $\mathcal{QD}$ obtained as the cross-product of the event spaces of queries and documents. Therefore, and this is Robertson's main claim, since Lafferty and Zhai do not apply the marginalisation equation to the full event space they may be easy pray of the marginalisation paradox.

The mismatch between event spaces appears to be a perfectly natural explanation for the lack of consistency between the results of applying marginalisation and those of direct frequency counting. In this particular case, however, the concept of event space is not derived from the applicability requirements of the marginalisation equation. Rather, less complete event spaces (e.g. $\mathcal{T}^+$ referred to in [6]) are identified *after* applying marginalisation in those cases where a paradoxical answer is claimed to be obtained. Event spaces are brought into life with the explicit purpose of resolving the marginalisation paradox and they are simply singled out by their being characteristic of two faulty applications of the marginalisation equation. Every time we find a minimum set of probability functions that, once employed in the marginalisation equation, let the results disagree with a frequentist interpretation of the same probabilities, we assign two different labels to the sets and we call the labels 'Event Space'.

2.2 Luk's Argument

Luk [5] understands the importance of the issue raised by Robertson, and he identifies the source of the problem mainly in the uniform probability assumption, which he considers not to bear any logical significance. The core issue is still how the probabilities that appear in the probabilistic model are coupled to worldly objects, but he shows to maintain that this link can be mapped, in addition to the event spaces, also onto the probability distributions: instead of letting the event space proliferate, we may as well keep the event space fixed and allow for multiple distributions. Luk shows that, in some configuration of distribution, the marginalisation paradox can be beaten and in this case rank equivalence holds, at least in a weak sense.

Luk's perspective specifies the distinction between the structural information carried by the event spaces, which can be graphically represented by the nodes of a tree structure and the information, which can be gathered about that structure and that is stored in the form of labels on the tree's branches (see Fig. 1 in Luk [5]). Given a direct probability measure onto the total event space, $\mathcal{ST}$ in Robertson's example, we can then construct an infinite number of distributions that result in the successful application of the marginalisation equation. Instead of focusing on one arbitrary configuration of distributions, e.g. a uniform probability distribution at each branch or, equivalently, at each marginalisation step, it is therefore far more instructive to assess, so goes Luk's argument, whether some distributions lead to data inconsistency, and avoid the paradox.

Luk follows a strategy which is somewhat anticipated in [1, pp. 1-10] where the authors make a distinction between rank equivalence at the probabilistic level and that at the lower statistical components level. Robertson only touched upon this distinction when trying to determine to which of the many possible event spaces are the different probabilistic models meant to be applied. One of the most interesting insights that Luk provides in [5] is indeed related to his appeal for a modular design of IR applications in which the probabilistic models are thought of as populating a probabilistic functional block that interacts to the various data sets through a statistical block. The major advantage of this scheme is that the effects, in this case on rank equivalence, of employing different probabilistic models can be kept separated from the contributions from the statistical components. The primitive interface to the data sets represented by the event spaces does not allow to easily determine how different informal presuppositions relate to the mathematical form of the probabilistic model; at this stage, it is not clear how to determine for a given IR model which event space it actually refers to.

In the next section we discuss what Luk seems reluctant to pursue as a consequence of his design, that is to allow for adding multiple blocks to the probabilistic model, which will inevitably loose some of its prominence. We maintain that the probabilistic model receives its empirical content from the statistical components model, admitting, however, that other functional blocks may also have ontological significance.

3 The Marginalisation Paradox and the Determination of the Priors

We attempt to untangle the complex, and only implicitly defined notion of event space given in [6], using what we call three different 'event space expansions'. The first expansion makes more explicit who makes an observation and assigns probabilities to uncertain knowledge. Probability theory is a faithful model of the real world only to the extent to which our knowledge about matter of facts is accurate, that is the uncertainty modeled by probability theory resides in our comprehension of the world and not in the world itself. A probabilistic model always refers, although sometimes implicitly, to an epistemic process. The latter can, if one wishes, be personified by a hypothetical observer who describes the world from her particular point of view. According to this picture we can recast Robertson's example by positing an observer within a universe with just two stars and three planets, who is puzzled by two apparently innocuous, but mutually exclusive statements that she believes both should be true. She finds out that measuring the magnetic field of planets that belong to stars and then separately that of the stars in order to subsequently infer by applying the marginalisation equation, the probability to find a planet with a magnetic field gives a certain result. She also believes that simply detecting and counting the number of planets with a magnetic field and then dividing the result by the total number of planets should yield the same number. A paradox arises if we want to maintain that both marginalisation and frequency counting are two legitimate ways to calculate probabilities, *and that they can be applied simultaneously.*

Our first claim is that the marginalisation paradox in [6] does not arise because of a shortcoming of probability theory that does not allow to adequately represent event spaces, but just because of the fact that the example presented by Robertson models an unrealistic process: in which realistic setting would an agent be interested into calculating $P(Y)$ by marginalisation when being in epistemic state $\mathcal{ST}$, which allows to simply count the occurrences of the Y variable?

We have thereby discovered a first link to elements of the real world, in this case agents characterised by certain epistemic states and processes in which they engage in order to modify their epistemic state, in a sense to evolve, which is curled into the notion of event space. The paradox reveals the presence of additional event spaces, besides the most intuitive full event space $\mathcal{ST}$, like $\mathcal{T}$ or $\mathcal{S}^+$ in [6], each labeling distinct epistemic states and processes.

This first expansion of event spaces in terms of an epistemic model resolves the paradox by inferring the presence of two different agents: one agent, being in the epistemic state $\mathcal{ST}$ calculates the probabilities by frequency counting; another agent, being in the lesser epistemic state $\mathcal{T}^+$ is forced to use marginalisation and she obtains results that disagree with the *other agent's* results. Needless to say, this happens all the time as agents may disagree on a lot of issues without generating paradoxes. Even stronger, the definition of agent as the collection of its information states, demands that agents cannot be in totally overlapping information states and still being distinguishable, hence no agent can totally agree with another agent.

This solution can also be explained in Luk's framework where an agent appears to use one IR model, while another uses a different model: in some configurations of the world the two agents may well derive the same conclusions, albeit when still in different epistemic states. This notion is indeed weaker than having one and the same agent using two models and ranking the documents in the same way, but it may be still worth investigating, as Luk does, which distributions lead to this result. The obvious advantage of the epistemic expansion, that we share with Luk's statistical expansion, is that we are able to distinguish the different gradations of rank equivalence and that we do not fall prey of the paradox; the advantage we have above Luk is that we are able to link the different cases to some procedural information, may that be made available by other components of the system like relevance feedback or other forms of user interaction.

Another expansion is particularly interesting for that it shows an unexpected connection between the marginalisation paradox in Robertson and another one, well known as the Monty Hall paradox [2]. One of its many versions involves three envelops, one of which contains a prize. After a contestant has chosen an envelop, the quiz-master opens one of the other two envelops, showing its empty content. She offers then the possibility to switch the envelops: the problem is to decide, based on objective reasons only, whether the contestant should accept the offer. At a first sight we may be tempted to reason like Lafferty and Zhai in [1, pp. 1-10], ignoring any event space and regarding relevance or the magnetic field as the prize. A naive conclusion would then be that, since the prize may be contained in either one or the other envelop, it does not really matter whether we switch or not; in fact, people that hear the paradox for the first time equally distribute their answers among the two possibilities. Once this metaphor is applied to IR, rank equivalence follows. The correct solution of the Monty Hall paradox is well known: the way we are supposed to calculate the probabilities depends on who put the prize in the envelop, the quiz-master or someone else. If the quiz-master knows which envelop contains the prize, and she would never open that one, her knowledge of the right envelope to open must be included in the model to arrive at the correct answer for the quiz candidate: switch!

We understand now what Robertson exactly wanted to model in his stars and planets example. Although it may seem strange that he models one observable with two different variables, the magnetic field with both X and Y, we encounter the same need in the Monty Hall paradox. In order to make sense of the situation we must model one observable, the prize, with two variables: one variable, say X, which models the magnetic field of the stars, but also the relevance of the queries or the prize that the quiz-master put in the envelop, and another variable, say Y, which models the magnetic field of the planets, but also the relevance of the documents or the prize that someone else than the quiz-master put into the envelop. Robertson, who claims to provide a model of the physical world, by choosing this representation for the magnetic field, ends up in representing some knowledge about the world that is not immediately evident upon examination of the data set, but is nevertheless needed for a correct determination of the probabilities; like the information on who put the prize in the envelop, which is not provided by the problem's statement, but it is exactly the source of the Monty Hall paradox.

We also immediately see how this analysis leads to another event space expansion. To specify the event space upon which probabilities are considered to apply, amounts then to the determination of the background information that should be taken into account in order to correctly calculate the probabilities. Also in this case the paradox does not sustain a more attentive analysis: there is no contradiction in the fact that a probabilistic reasoning moving from two different sets of background information also leads to two different results. Discovering some local regularity is actually a quite interesting finding and therefore Luk's weak equivalence should not be underestimated; he did however fail to recognise that it is not the case that there are either different event spaces or different possible distributions, but there are different problems, for different observers, with different prior information, but possibly on the same data set. The event spaces show yet another face by playing the role, which the priors in Bayesian reasoning are usually charged with: they appear to be the core issue in the paradox resolution only because they are a primitive description of the prior information, which *de facto* resolves the paradox.

4 Drawing Conclusions and Consequences for IR

In this paper we show that the way Robertson in [6] questions the rank equivalence derived by Lafferty and Zhai in [1, pp. 1-10] is ineffective. The alleged paradox that would arise when the event spaces are not adequately taken into account, can be defeated in more than one way, each corresponding to one of what we termed ‘event space expansions’. Luk has proposed to extend the event space, understood as an interface with the data set only, by means of a statistical components module. While we welcome his attempt towards a more thoroughly understanding of the issues raised by Robertson by means of a functional hierarchy, we have argued that his claim is in itself also problematic because it does not address the most sensitive features of Robertson’s argument: the existence of the paradoxical situation and the functional identity between the priors and the event spaces. Luk descends the hierarchy and attempts to solve this issue at the statistical level. We view the problem as one that needs more abstraction rather than less; if event spaces are really just priors, what we need is a method to select priors, which become sockets to interface upper-level functional blocks. For example, a probabilistic model can fetch, through its priors, the output of a dynamic epistemic logic module that formalises how observers change their information states upon which their relevance assessments highly depend.

Once we apply this conclusion to IR as a discipline, this is in essence an argument for revisiting the logical models of IR as first proposed in [7] for the solution to the paradox that has arisen calls an higher level of abstraction than that provided either by the probabilistic model or by the statistical components model.

References

1. Croft, B.W., Lafferty, J.: Language Modeling for Information Retrieval. The Information Retrieval Series. Springer, Heidelberg (1999)
2. Gardner, M.: Mathematical games column. In: Scientific American, October 1959, pp. 180–182 (1959)
3. Hiemstra, D.: Using language models for information retrieval. Ph.D. dissertation. University of Twente, Enschede (January 2001)
4. Jones, K.S., Walker, S., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments. Inf. Process. Manage. 36(6), 779–808 (2000)
5. Luk, R.W.: On event space and rank equivalence between probabilistic retrieval models. Inf. Retr. 11(6), 539–561 (2008), http://dx.doi.org/10.1007/s10791-008-9062-z
6. Robertson, S.: On event spaces and probabilistic models in information retrieval. Inf. Retr. 8(2), 319–329 (2005), http://dx.doi.org/10.1007/s10791-005-5665-9
7. van Rijsbergen, C.J.: A new theoretical framework for information retrieval. SIGIR Forum. 21(1-2), 23–29 (1987)

Robust Word Similarity Estimation Using Perturbation Kernels

Kevyn Collins-Thompson

Microsoft Research
1 Microsoft Way
Redmond, WA 98052
kevynct@microsoft.com

Abstract. We introduce *perturbation kernels*, a new class of similarity measure for information retrieval that casts word similarity in terms of multi-task learning. Perturbation kernels model uncertainty in the user's query by choosing a small number of variations in the relative weights of the query terms to build a more complete picture of the query context, which is then used to compute a form of *expected distance* between words. Our approach has a principled mathematical foundation, a simple analytical form, and makes few assumptions about the underlying retrieval model, making it easy to apply in a broad family of existing query expansion and model estimation algorithms.

1 Introduction

A fundamental research problem of information retrieval is how to improve search effectiveness by learning an extended representation of the user's information need, called a *query model*, that captures more about the context of an information need than is available from the few words in the query itself. For example, in performing a type of query expansion, a very simple query model might take the form of a unigram language model over words related to the user's query terms. One significant problem in performing query expansion is the *risk* of adding words that are unrelated to the query, causing the query model to 'drift' away from the user's original intent. Thus, improving the quality and reliability of the similarity measure used to find related terms is an important goal in itself.

With this problem in mind, we introduce *perturbation kernels*, which cast estimating word similarity as an type of multi-task learning problem. Informally, the key idea of perturbation-based kernels is that two input objects x and y, such as words, are considered similar in the context of a given query Q if probability distributions $p(x|Q)$ and $p(y|Q)$ that depend on Q are affected in similar ways with small variations in Q.

Our approach has several advantages. First, the use of query perturbations results in sensitivity features that give more precise word similarity relations, which in turn can improve the stability of query expansion algorithms that use them. Second, we make few assumptions about the nature of the underlying retrieval model, meaning that such similarity measures may be applied in a

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 265–272, 2009.

wide variety of existing query model estimation or expansion methods. Third, our solution has sound theoretical justification with close connections to kernel methods such as the leave-one-out kernel [13], robust approximation, and metric learning. Finally, our algorithm has a simple, efficient analytical form.

2 Mathematical Formulation

Let $\mathcal{X}$ be the input domain of interest (e.g. words) from which a training example (query) is generated. For any $x \in \mathcal{X}$ we identify an m-dimensional vector called a *feature mapping*, denoted $\phi : \mathcal{X} \rightarrow \mathbb{R}^m$. With this feature mapping, we define a symmetric kernel function $k(\cdot)$ to measure the closeness of input points x and y as $k(x, y) = \phi(x) \cdot \phi(y)$ where the right side of the equation is the *inner product* of $\phi(x)$ and $\phi(y)$.

A *perturbation* to a training set of n instances $x = \{x_1 \ldots x_n\}$ can modeled by a vector of counts $\alpha = \{\alpha_1, \ldots, \alpha_n\}$ with count α_i corresponding to the weight of training example x_i. For the original training set, $\alpha_i = 1$ for all instances x_i. To leave out the instance x_i, we set $\alpha_i = 0$. To give x_i more weight, we set $\alpha_i > 1$. A *perturbation strategy* is a set $\mathcal{A} = \{\alpha_i\}$ of perturbation vectors. The set $\mathcal{A}$ may be selected with either a random or deterministic process. We use the following deterministic perturbation strategies in this study to define uncertainty sets around the initial query Q:

- The *leave-one-out* strategy (LOO) has $\mathcal{A} = \{\alpha_1, \ldots, \alpha_n\}$ where $\alpha_i[j] = 0$ for $i = j$ and 1 otherwise.
- The *term-at-a-time* strategy (TAT) is complementary to LOO and uses $\mathcal{A} = \{\alpha_1, \ldots, \alpha_n\}$ where $\alpha_i[j] = 1$ for $i = j$ and 0 otherwise.

These strategies are extremely simple to implement, widely used for query expansion and performance prediction tasks, and fast to execute in a real-time search environment. They both use $N + 1$ variants of the query, while being somewhat complementary strategies, making them ideal for comparison. The TAT and LOO methods are different extremes in a more general class of combinatorial methods that could be defined on the set of query terms. We leave exploration of more sophisticated perturbation schemes for future work.

We denote the probability distribution of $x \in \mathcal{X}$ that results from a perturbation α_i as $p^{(i)}(x)$. In the context of information retrieval, we view a query q as a training set of n instances of query terms $\{q_1, \ldots, q_n\}$ where q_i is drawn from $\mathcal{X} = \mathcal{V}$ for vocabulary $\mathcal{V}$. In the next sections, we derive the general form of the perturbation kernel and feature mapping $\phi(x)$.

2.1 Canonical Similarity Integrals

Our formulation of perturbation-based similarity is inspired mainly by earlier work of Baxter using auxiliary tasks for classification and function approximation, as well as a more recent followup article by Minka [11] that discussed distance measures as prior probabilities. Baxter showed that for 1-nearest-neighbor classification, there is a unique optimal similarity measure that he called the Canonical Distortion Measure (CDM) [3]. In the classification setting, this quantity $\delta(x_1, x_2)$ is the expected loss of classifying x_1 with x_2's label. The expectation is taken over a probability space of classifiers (tasks). To apply Baxter's idea to information retrieval applications, we view a task as relevance estimation with respect to a particular query q. We call a task distribution for q a *query neighborhood* of q. One way to define the query neighborhood is as a probability measure $Q(f)$ over task functions $f_q(x)$ where $f_q(x)$ gives a 'soft' label $p_q(x|\theta_R)$ with respect to the unknown 'true' Relevance Model θ_R for query q. The canonical similarity measure $\varDelta_q(x, y)$ is then the expected loss over Q, given x and y in the input domain $\mathcal{X}$.

$$\varDelta_q(x, y) = \int_{\mathcal{F}} \rho(f(x), f(y)) dQ(f) \tag{1}$$

This measure is uniquely determined by the task function f, the choice of query neighborhood measure Q and loss function $\rho(u, v)$. We fix the two input domain elements x and y (words) and integrate over a probability space $\mathcal{P}$ of density functions. These density functions $p^{(\alpha)}(\cdot)$ are those that result from perturbations α on the training data (query), and we assume we have a measure $Q(p)$ over $\mathcal{P}$ that describes the distribution over perturbation densities $p^{(\alpha)}(\cdot)$. We assume that the density $p^{(\alpha)}(\cdot)$ is defined for all elements of the input domain $\mathcal{X}$ (although it may be zero), so that the integral exists for all pairs $(u, v) \in \mathcal{X}^2$ with measure $G(u, v|x, y)$. The general form of the perturbation kernel is then defined to be an expected distance between x and y:

$$k_q(x, y) = \int_{\mathcal{X}^2} \int_{\mathcal{P}} \rho(p^{(\alpha)}(u), p^{(\alpha)}(v)) dG(u, v|x, y) dQ(p). \tag{2}$$

One natural choice for $\rho(x, y)$ is $\rho(x, y) = (\sqrt{x} - \sqrt{\kappa})(\sqrt{y} - \sqrt{\kappa})$ for a fixed origin κ since this converts the integral in Eq. 2 to a form of Hellinger inner product between vectors $\{p^{(\alpha)}(x)\}$ and $\{p^{(\alpha)}(y)\}$ when we set $\kappa = p(x)$. This choice is motivated by connections with Fisher kernels [13], but other choices for $\rho(x, y)$ are open to exploration (as well as other perturbation strategies).

2.2 Approximating the Similarity Integral

By writing the similarity measure in Eq. 2 as an integral we can bring to bear general-purpose integration methods. Recall that the basic approach to evaluate a general integral of the form $\mathcal{I} = \int_{\varTheta} f(\theta) d\mu(\theta)$ on the domain $\varTheta$ with measure $d\mu$ is to independently sample N points $X_1, \ldots X_N$ in $\varTheta$ according to some density function $p(x)$, and then compute the random variable $F_N = \frac{1}{n} \sum_{i=1}^{n} \frac{f(X_i)}{p(X_i)}$. A simple importance sampling approximation to Eq. 2 can be derived by assigning the perturbation densities $\{p^{(\alpha)}(x)\}$ equal probability, and taking a single-sample approximation at (x, y) in the $\mathcal{X}^2$ domain, where we assume there is some sampling distribution MLE $\hat{p}(x,y)$ and that words x and y are independent (given relevance) so that $\hat{p}(x,y) = \hat{p}(x) \cdot \hat{p}(y)$ for MLE $\hat{p}(x)$ over $\mathcal{X}$, giving

$$k_q(x,y) \approx \frac{1}{\hat{p}(x,y)} \cdot \frac{1}{n} \sum_{i=1}^{n} \rho(p^{(i)}(x), p^{(i)}(y)) = \sum_{i=1}^{n} \phi(x_i)\phi(y_i) \tag{3}$$

so that the feature mapping vector $\phi(x)$ has entries

$$\phi_i(x) = \frac{1}{\sqrt{n}} \frac{\sqrt{\hat{p}^{(i)}}(x) - \sqrt{\hat{p}(x)}}{\hat{p}(x)}. \tag{4}$$

Typically, $\hat{p}(x)$ represents the distribution obtained using the results from the initial query, which can be considered a null perturbation.

2.3 Algorithm

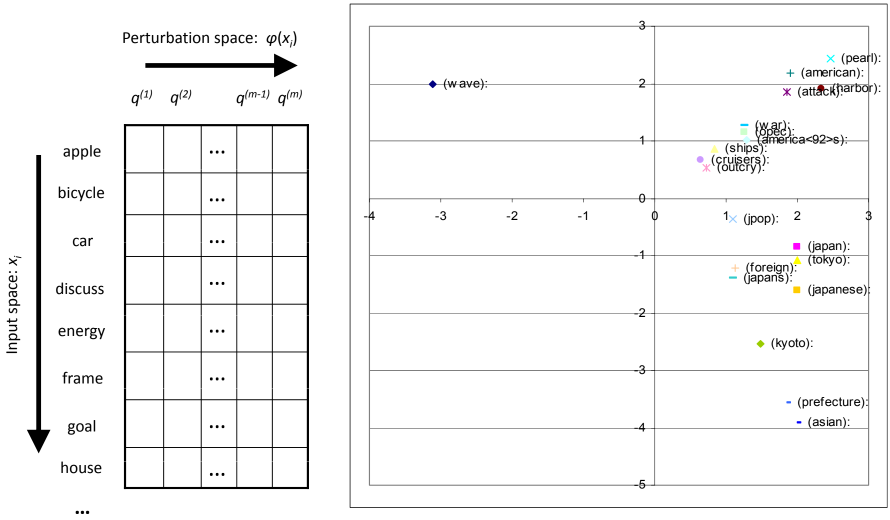
The specific steps of the perturbation kernel algorithm are given in Fig. 1. Here we show the case where we use the language modeling approach to IR, and estimate a unigram Relevance Model [10] $p_q(w|R)$ for a given query q and word $w \in \mathcal{V}$. The resulting distance matrix D, which can be viewed as a graph over words, is then typically used as one of the inputs to a specific query model estimation or expansion algorithm.

1. Given initial query q, generate N query perturbations $q^{(i)}$, $i = 1 \ldots N$.
2. Run q to generate a corresponding Relevance Model $\hat{p}(w) = p(w|R)$, and similarly run each $q^{(i)}$ to generate Relevance Model $p^{(i)}(w|R)$.
3. Compute matrix A, which has one row $\phi(w)$ for each word w in vocabulary $\mathcal{V}$, with the i-th entry $\phi_i(w)$ of the row computed using Eq. 4.
4. Compute the final word-word distance matrix D having entry $d_{ij} = g_{ii} + g_{jj} - 2g_{ij}$ where g_{ij} are the entries of the Gram matrix $G = AA^{\mathsf{T}}$.

Fig. 1. The perturbation kernel algorithm (Relevance Models)

2.4 Visualizing Perturbation Similarity

Plotting words in feature mapping (perturbation) coordinates shows useful local and global distance properties. An example is shown in Figure 2. The x-axis plots $\log \phi_1(w)$ and the y-axis plots $\log \phi_2(w)$ where the feature mapping $\phi(w)$ is given in Eq. 4, and the perturbation strategy uses LOO query variants. Words with similar relative changes in relevance model $p(w|R)$ to the same query perturbations are close in this space. The mutual proximity of the query terms gives an indication for how phrase-like their behavior is, while the global position of clusters from the origin is related to their relevance to the query. Here, the 'japanese wave' query seeks information about tsunamis; the irrelevant 'Pearl Harbor' noise cluster has been successfully separated from the other terms and placed in the NE quadrant, while the SE quadrant brings together words much more closely related to 'japanese', such as 'asian' and 'prefecture'.



(a) Perturbation matrix A

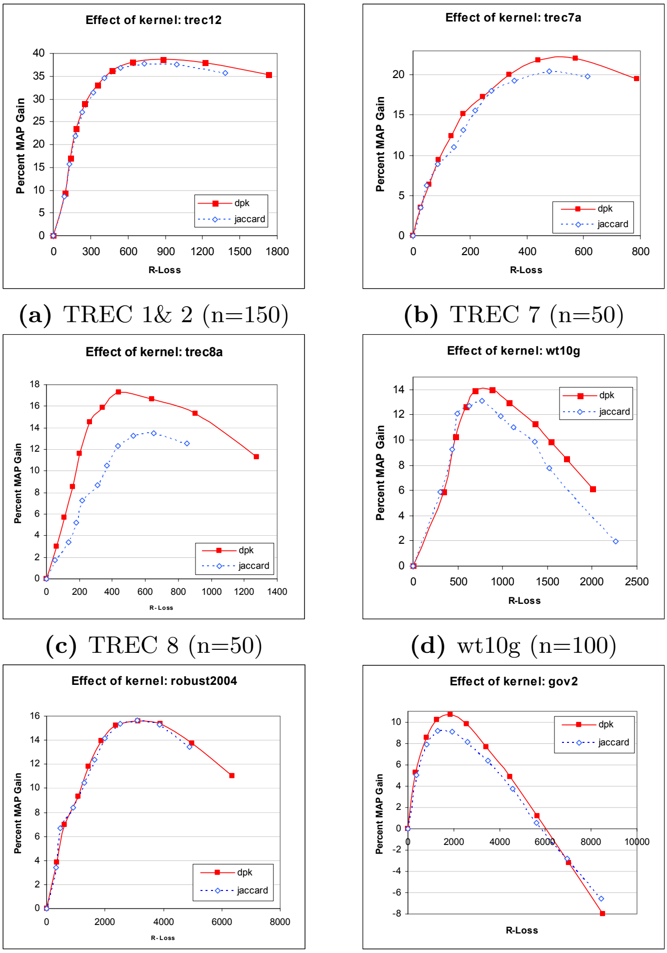
(b) Visualization example

Fig. 2. (a) The matrix of probability vectors for a discrete input space (here representing a word vocabulary). Each column represents the discrete parametric or non-parametric probability distribution across all words estimated from a particular perturbation of the training data (i.e. query). The rows $\phi(x_i)$ give the probability estimates across all perturbations for a given word x_i. (b) Visualization showing how the perturbation kernel is effective at term clustering for the top 20 expansion terms for TREC topic 491 'japanese wave'. The first two co-ordinates in perturbation space are plotted as x and y axes. Terms whose probabilities respond similarly to the same query perturbations are close in this space. Close words have been jittered apart for clarity.

3 Evaluation for Query Expansion

Since our focus here is on making expansion algorithms more stable, we introduce *risk-reward tradeoff curves* to visualize and compare the *risk profile* of query expansion algorithms [5]. The x-axis gives a measure of downside risk or variance of the expansion algorithm, by counting the net loss of relevant documents for queries whose initial results are made worse by using the expansion algorithm (which we call R-Loss). The y-axis gives the actual relative gain or loss from using expansion, as measured by Mean Average Precision (MAP) gain over all queries. The eleven points on each curve show how the risk-reward tradeoff changes as the query model for the initial query is interpolated using parameter α with the expansion model: from $\alpha = 0$ at the origin (initial query only) to $\alpha = 1.0$, where the query model consists entirely of the expansion model. The curves for six standard TREC topic sets are shown in Figure 3, with the number of queries in the topic set given in parentheses.

The risk-aware query expansion framework we use is described in detail in [4]. For a baseline word similarity measure, we wanted a method that could be



(a) TREC 1& 2 (n=150) **(b)** TREC 7 (n=50)

(c) TREC 8 (n=50) **(d)** wt10g (n=100)

(e) Robust 2004 (n=250) **(f)** gov2 (2004-2006, n=150)

Fig. 3. Risk-reward tradeoff curves for six TREC topic sets, showing how the perturbation kernel can improve the risk profile of an expansion algorithm. The solid line is the curve given by the expansion algorithm using the perturbation kernel. The dashed line uses the same expansion algorithm and parameter settings, but substitutes a Jaccard kernel that does not use sensitivity information. Tradeoff curves that are *higher and to the left* give a better risk-reward tradeoff. Curves are plotted with points at α-increments of 0.1, starting with $\alpha = 0$ at the origin and increasing to $\alpha = 1.0$.

calculated from just the initial set of top-retrieved documents. Recall that we are deriving term association statistics from a set of documents that is already biased toward the query terms, so that the number of documents *not* containing a query term is frequently zero, or close to zero. We chose the Jaccard measure for this study since it is a simple, widely-known term association measure that ignores this non-relevant negative information.

For four of the six collections (TREC 7, TREC 8, wt10g, and gov2) the perturbation kernel improved the risk profile of the query expansion algorithm, particularly in the typical operational zone from $\alpha = 0$ to $\alpha = 0.5$. At a setting of $\alpha = 0.5$, the improvements are largest for TREC 8 and gov2. For TREC 8, the perturbation kernel gives a MAP gain of 14.5% with R-Loss of 262, while the Jaccard kernel gives a MAP gain of 8.68% with R-Loss of 307. For the gov2 collection, the perturbation kernel MAP gain is 9.78% with R-Loss of 2555, while the Jaccard kernel has MAP gain of 8.13% with R-Loss of 2605. For two of the collections (TREC 1&2 and Robust 2004), the performance of the two kernels is almost identical: TREC 1&2 shows only a tiny advantage for the perturbation kernel for $\alpha \geq 0.6$. We found that LOO perturbation had a small but consistently dominance in performance over TAT for all experiments and so we report only LOO experiments here. The results suggest that the perturbation kernel gives the potential for useful gains on some collections, with little downside risk.

4 Related Work

Recent research on kernels has developed a broad family based on inner products over probability distributions. When we assign each input point x_i a probability distribution over input space, we can integrate over input space – in the discrete case, the *columns* of A shown in Figure 2a, instead of the *rows* of A. This type of similarity measure includes probability product kernels [8]; the leave-one-out (LOO) kernel [13]; and marginalized kernels [14]. Fisher kernels [7], a special case of marginalized kernel, compare the sufficient statistics of generative models and are well-suited to query model problems, because they can exploit unlabeled data: the similarity of two data items is not only a function of the items themselves but also their context. There are also interesting connections to information diffusion kernels [9] and statistical translation models such as those developed by Dillon *et al.* [6], where word similarity is defined in terms of the probability that two words have the same context.

In multi-task learning, in addition to Baxter [3], Ando *et al.* used a multi-task learning framework [2] in a preliminary TREC genomics study [1] where they noted the connection between multi-task learning and using auxiliary queries. For Web retrieval, Sahami and Heilman [12] proposed a kernel for comparing text snippets using the inner product of the query expansions that result by considering each text snippet as a Web query. None of these methods, however, explored the use of sensitivity information or framed the similarity problem in terms of multi-task learning to improve stability of the 'client' algorithm.

5 Conclusions

The perturbation kernel is a useful tool for comparing the similarity of elements in a domain $\mathcal{X}$, such as words, when we have a probability distribution over $\mathcal{X}$ whose functional form may be unknown and/or highly complex and which is estimated based on a very small training set over elements from $\mathcal{X}$. Similarity between elements is induced with respect to small perturbations in the training data, so that each input point is identified with multiple probability densities evaluated at that point that are integrated over probability density space. We showed how casting word similarity estimation as a multi-task learning problem of this type can exploit knowledge from multiple 'tasks' in the form of query perturbations. Our initial evaluation suggests that the perturbation kernel is a more stable replacement for similar baseline measures that ignore sensitivity. More generally, it represents a step in a fruitful research direction: exploring how information retrieval algorithms can exploit higher-level risk or variance information to improve their performance [5]. Further improvements may be possible with more sophisticated perturbation strategies for the query, such as those learned from Web query logs or user profiles.

References

1. Ando, R.K., Dredze, M., Zhang, T.: TREC 2005 genomics track experiments at IBM Watson. In: Proceedings of TREC 2005, NIST Special Publication (2006)
2. Ando, R.K., Zhang, T.: A framework for learning predictive structures from multiple tasks and unlabeled data. J. Mach. Learning Research 6, 1817–1853 (2005)
3. Baxter, J.: The canonical distortion measure for vector quantization and function approximation. In: ICML 1997, pp. 39–47 (1997)
4. Collins-Thompson, K.: Estimating robust query models using convex optimization. In: Advances in Neural Information Processing Systems (NIPS), vol. 21, pp. 329–336. MIT Press, Cambridge (2008)
5. Collins-Thompson, K.: Robust Model Estimation Methods for Information Retrieval, PhD thesis. Carnegie Mellon University (2008)
6. Dillon, J., Mao, Y., Lebanon, G., Zhang, J.: Statistical translation, heat kernels, and expected distances. In: UAI 2007, pp. 93–100 (2007)
7. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: Advances in Neural Information Processing Systems(NIPS), vol. 11, pp. 487–493. MIT Press, Cambridge (1999)
8. Jebara, T., Kondor, R., Howard, A.: Probability product kernels. J. Machine Learning Research 5, 819–844 (2004)
9. Lafferty, J.D., Lebanon, G.: Information diffusion kernels. In: Advances in Neural Information Processing Systems (NIPS), vol. 15, pp. 375–382. MIT Press, Cambridge (2002)
10. Lavrenko, V.: A Generative Theory of Relevance. PhD thesis, Univ. of Massachusetts, Amherst (2004)
11. Minka, T.: Distance measures as prior probabilities. Technical report (2000)
12. Sahami, M., Heilman, T.: A web-based kernel function for measuring the similarity of short text snippets. In: Proc. of WWW 2006, pp. 377–386 (2006)
13. Tsuda, K., Kawanabe, M.: The leave-one-out kernel. In: Dorronsoro, J.R. (ed.) ICANN 2002. LNCS, vol. 2415, pp. 727–732. Springer, Heidelberg (2002)
14. Tsuda, K., Kin, T., Asai, K.: Marginalized kernels for biological sequences. Bioinformatics 18(1), 268–275 (2002)

Possibilistic Similarity Estimation and Visualization

Anas Dahabiah, John Puentes, and Basel Solaiman

TELECOM Bretagne, Image and Information Processing Department, Technopôle
Brest-Iroise, CS 83818, 29238 Brest Cedex 3, France

Abstract. In this paper, we present a very general and powerful approach to represent and to visualize the similarity between the objects that contain heterogeneous, imperfect and missing attributes in order to easily achieve efficient analysis and retrieval of information by organizing and gathering these objects into meaningful groups. Our method is essentially based on possibility theory to estimate the similarity and on the spatial, the graphical, and the clustering-based representational models to visualize and represent its structure. Our approach will be applied to a real digestive image database (http://i3se009d.univ-brest.fr/ password view2006 [4]). Without any a priori medical knowledge concerning the key attributes of the pathologies, and without any complicated preprocessing of the imperfect data, results show that we are capable to visualize and to organize the different categories of the digestive pathologies. These results were validated by the doctor.

Keywords: Similarity, Possibility Theory, Graph Theory, Scaling, Clustering.

1 Information Imperfection and Heterogeneity Visualization Problematic

The word *data* is the Latin plural of *datum*, the past participle of the verb *dare* (to give), hence data mean givens (any raw symbols, numbers, words, images, measurements, inputs, etc. that exist but have no significance or meaning beyond their existence). Giving more meanings and descriptions by means of relational connections can transform data into information. For instance, the datum "37" given as a number can be transformed into information by adding some other elements like "37 °C" which informs us that this measure represents a temperature, and by telling which type of temperature we describe (human or ambient temperature for example). Consequently, information can be defined as a function called the informative function described by means of a model that maps the information definition set that represents the object of our description to the information content set that represents the manner used to describe the information (see Fig. 1(a)) [8].

Any information can be characterized by three main indicators [1]: the exhaustiveness, the exclusiveness, and the confidence degree (linkage strength).

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 273–280, 2009.

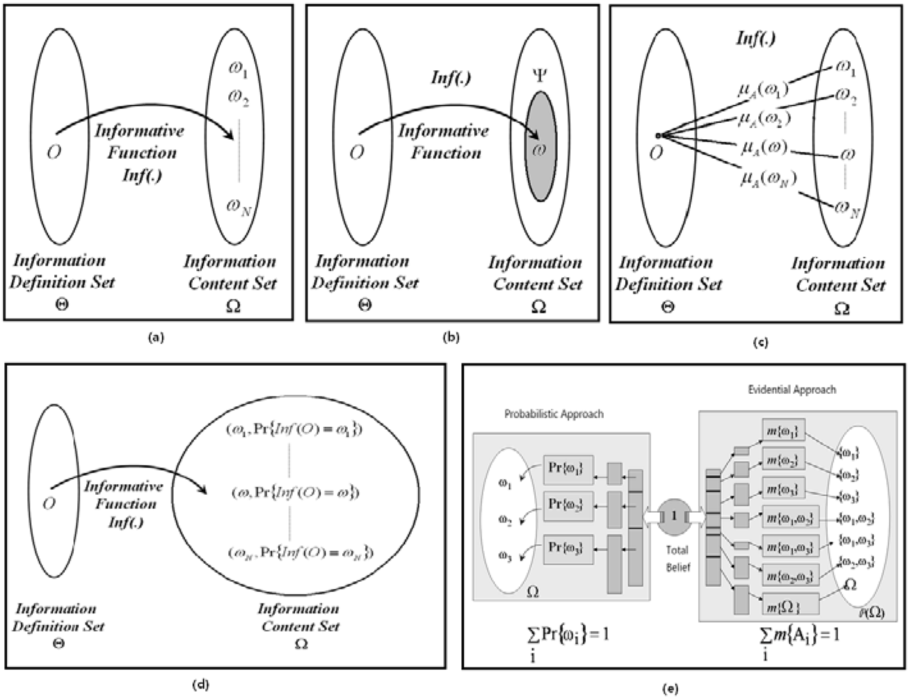


Fig. 1. (a) General scheme of information structure, (b) an imprecise information element, (c) an ambiguous information element, (d) a probabilistic information element, (e) the difference between the probabilistic and the evidential approach

We say that the information is exhaustive if $\forall O \in \Theta$, we can find an element ω in Ω such that $Inf(O) = \omega$. This means that Ω contains all the possible hypotheses, classes, decisions, labels, description, etc. We consider that the information is exclusive if there is only and only one element ω in Ω such that $Inf(O) = \omega$. Concerning the third property, information can have either binary linkage strength (the linkage $Inf(O) = \omega$ is completely true or completely false) or partial linkage strength (the linkage $Inf(O) = \omega$ is associated with a weight, a membership degree, etc.). According to theses indicators (or characteristics), we have two mathematical models used to deal with the imperfection of information. The first one is called "probabilistic uncertainty" which concerns the exhaustive and exclusive information elements with total confidence degree in which the knowledge regarding the identity of ω in $(Inf(O) = \omega)$, $O \in \Theta$ is described: (i) either via a subset $\Psi \subseteq \Omega$, as is shown in Fig. 1(b) (imprecise information); (ii) or as a probability distribution defined over Ω, as illustrated in Fig. 1(d) (probabilistic information); (iii) or by means of belief masses defined over the subsets of Ω rather than over the elements of Ω (the simple hypothesis or the singletons) as depicted in Fig. 1(e) (evidential information); (iv) or finally via an ambiguous knowledge given as linguistic variables or as possibility distributions (possibilistic information). Our objective in the probabilistic uncertainty models is to look for the only (unknown) $\omega \in \Omega$ using probability, evidence (Dempster-Shafer), or possibility theory [8]. The second model used in information imperfection processing is called "the ambiguity". As its name implies, this model can be exploited to deal with ambiguous information elements which are exhaustive, but not necessarily exclusive, since each object $O \in \Theta$ can be associated with several informational contents (several elements) of Ω with the same or with different degrees of strength called the membership degrees (denoted μ in Fig. 1(c)). Our objective in this model is to combine several criteria or alternatives at the same time in order to take a decision using the fuzzy set theory. In addition to the different types of information element imperfection, the informational contents can be quantitative (numerical or binary), or qualitative (nominal and ordinal) regarding the measuring scale of information elements. The main question that arises in this context is how to model the structural representation and how to visualize the relationships and the resemblance between objects (records, cases, etc.) having different kinds of imperfect information elements (imprecise, probabilistic, evidential, possibilistic, ambiguous, or even missing data) besides various types of measuring scales (quantitative and qualitative) in a unified framework, by taking advantages of all the conventional visualization and clustering well-developed approaches without adding a significant modification and without increasing the execution computation time. In the literature there are some humble and simple attempts to tackle some aspects of these concepts, like handling the imprecise nominal information elements without taking account of the other types of imperfection by using a method that depends on several parameters whose calculation is somehow empirical and implicit as in [6], or as another example the recent works of Zemerline [10] that outperforms many prior works in handling the heterogeneous (numerical and nominal) information thanks to the fuzzy set theory, neglecting however the ordinal and the imperfect information, and using an approach overburdened with lots of constraints and conditions. Herein, we propose an approach that takes account of all the aforementioned points to measure the similarity (section 2) and to visualize it using the conventional representational models (section 3) using the possibility theory which is situated at the confluence of all the other mathematical models that tackle information imperfection [5]-[1], and which is capable to transform the informational contents from all measuring scales to possibility degrees taking account of the heterogeneity of information at the same time. Our experimental results are represented in section 4 and discussed in section 5.

2 Possibility-Based Similarity Measuring

According to possibility theory, we can estimate the veracity of or the matching between a fuzzy proposition "V *is* A" defined via $\mu_A(\omega)$, given a referential fuzzy proposition "*V is B*" defined via $\mu_B(\omega)$ (where V stands for variable and A, B are the informational contents) using the possibility measure $\Pi(A,B)$, and the necessity measure $N(A,B)$, defined as [2]:

$$\Pi(A,B) = sup_{\omega\in\Omega} min(\mu_A(\omega), \mu_B(\omega)), \tag{1}$$

$$N(A,B) = inf_{\omega\in\Omega} max(\mu_A(\omega), 1-\mu_B(\omega)). \tag{2}$$

These two equations constitute the backbone of the proposed possibilistic similarity model. Let us suppose now that we want to measure the similarity between two objects characterized by their S-dimensional descriptor vectors $A_j = [a_{1j}, a_{2j}, ..., a_{ij}, ..., a_{Sj}]$ and $A_k = [a_{1k}, a_{2k}, ..., a_{ik}, ..., a_{Sk}]$. Unlike the conventional vectors, these vectors may contain imperfect and heterogeneous information elements. To take account of the viewpoint of the expert and in order to personalize this process when it is necessary, we suppose that each attribute is associated with a "tolerance function" defined by an expert as a formula or as a table permitting to describe mathematically to which degree we consider that two values of this attribute are similar [4].

In our approach the similarity between the two objects characterized via A_j and A_k (called inter-object similarity) can be estimated by means of two measures: the possibility degree of similarity that tells us to which degree it is possible that these vectors are similar, and the necessity degree of similarity of these vectors that tells us to which degree we are certain of their similarity. To calculate these degrees we must firstly calculate the inter-attribute possibility and necessity degrees and aggregate them by taking their average for example. Accordingly, for each attribute, we consider that its associated tolerance function (denoted μ_a) is the fuzzy proposition, that must be matched with the compound referential fuzzy proposition given as the informational content of the attribute a_{ij} in A_j is defined via the possibility distribution $\pi_{A_j,a_{ij}}(\omega)$ and the informational content of the attribute a_{ik} in A_k is defined via the possibility distribution $\pi_{A_k,a_{ik}}(\omega)$ that can be mathematically defined as:

$$\pi_\Omega(a_{ij}, a_{ik}) = min\left(\pi_{A_j,a_{ij}}(\omega), \pi_{A_k,a_{ik}}(\omega)\right). \tag{3}$$

In this case we can calculate inter-attribute possibility and necessity degrees of similarity from the two basic equations introduced at the beginning of this section as:

$$\Pi_i(a_{ij}, a_{ik}) = sup_{u\in U}\left[min(\mu_a(u), \pi_\Omega(u))\right], \tag{4}$$

$$N_i(a_{ij}, a_{ik}) = inf_{u\in U}\left[max(\mu_a(u), 1-\pi_\Omega(u))\right], \tag{5}$$

where $U = \Omega \times \Omega$.

We consider that if the value of an attribute is given in one object and is unassigned in the other (the case of missing values), it is completely possible that these values are similar $\Pi_i = 1$ but we are entirely uncertain $N_i = 0$.

3 Similarity Representational Models

Visualization is the process of transforming invisible abstract data, information, and knowledge into a visible display in the form of geometric or graphical representations in order to support tasks such as data analysis, information exploration, trend prediction, pattern detection, rhythm discovery and so. Actually, these representational models give observed events a meaningful interpretation and allow future or unseen events to be anticipated through the process of generalization [9]. In order to represent the similarity, we have chosen well-developed mathematical representational models like the linear and the circular unidimensional scaling (LUS and CUS) [7] and the multidimensional scaling models (MDS) [9] for similarity spatial visualization, the additive and the ultrametric trees [1],[9] for the graph-based visualization, and the evidential, hierarchical, and additive clustering [9],[3] for clustering-based visualization. The spatial models locate each object in a multidimensional coordinate space (along a linear or around a closed circular continuum), and assume that the similarity between any two objects is a function of how close they are to one another. Tree models represent objects as the terminal nodes in an acyclic graph, in such a way that the similarity between two objects is considered to be inversely related to the length of the unique path that connects them. Regarding the structural models, the similarity between any two objects can be represented as belief masses in the evidential clustering (based on the fact that information conflict degree between the belief masses of any two objects reflects their dissimilarity), as the length of the path that connects them in the dendrogram in the hierarchical clustering, or as the weight of their shared sets in the additive clustering. It is not the intent of this article to present formal demonstrations of these structural representations. All of this is generally available in details in the literature that we indicated. The primary interest here is to show for the first time how these models can be applied to a possibilistic similarity matrix giving robust results in the presence of heterogeneous, imperfect, and even missing information elements.

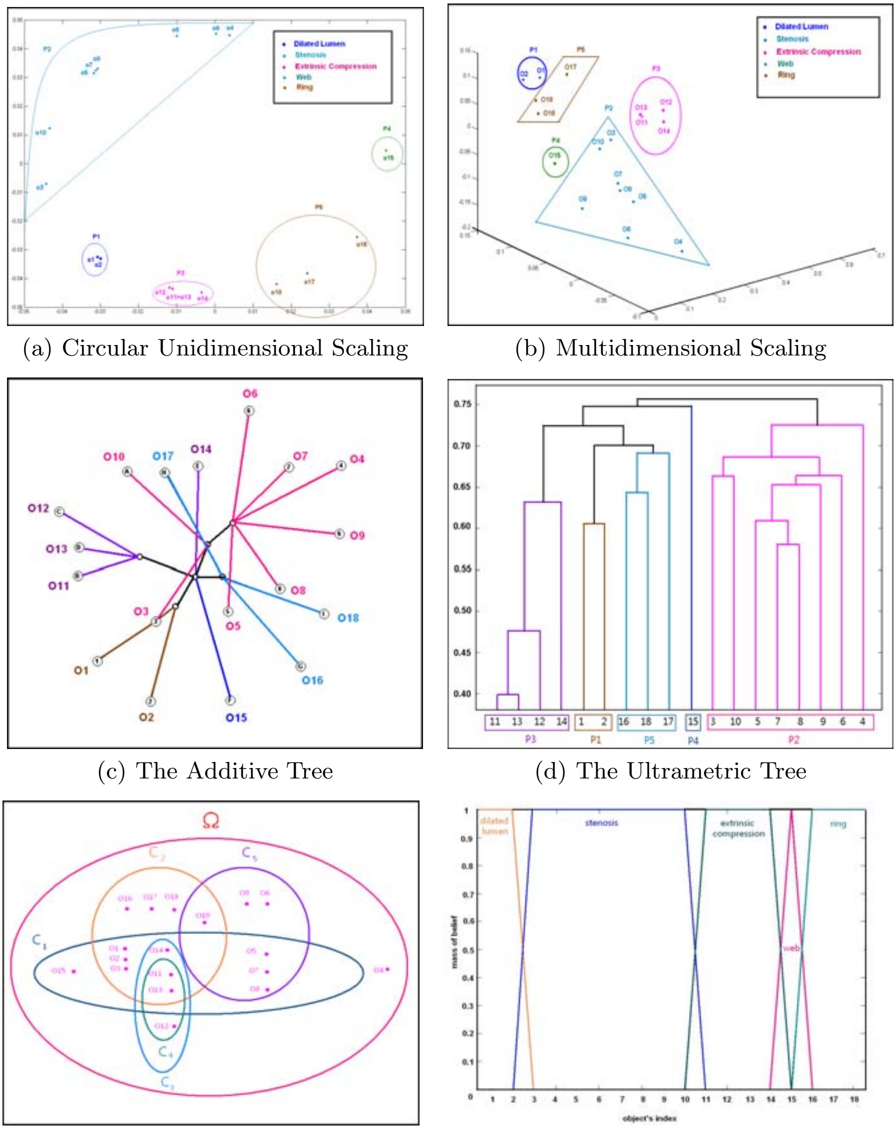
4 Experimental Study

The digestive endoscopic database that we used in this paper is well described in [4]. It consists of objects (images) described by an expert (33 attributes with 206 modalities). We will show in the following that the graphical representations of the similarity necessity matrix of the database are capable to obviously show the classes of the pathologies of the images. In order to have a simple and a clear representation of our results, we will show in the following as an example a small subset of pathologies belonging to our global data set, keeping in mind that this analysis is applicable to any other case because the approach is general and the matrices that we use are submatrices of the general necessity matrix applied to all the objects of our global data set.

Suppose that: $CB = \{O_1, O_2, ..., O_{18}\}$ is a casebase where $P_1 = \{O_1, O_2\}$ is the set of the objects whose pathology class is "Dilated Lumen", $P_2 = \{O_3, O_4, O_5, O_6, O_7, O_8, O_9, O_{10}\}$ is the set of the objects whose pathology class is “Stenosis (esophagus)”, $P_3 = \{O_{11}, O_{12}, O_{13}, O_{14}\}$ is the set of the objects whose pathology class is “Extrinsic Compression”. O_{11} and O_{13} are very similar, $P_4 = \{O_{15}\}$ is the set of the object whose pathology class is “Web-shape”, and $P_5 = \{O_{16}, O_{17}, O_{18}\}$ is the set of the objects whose pathology class is “Ring-shape” (The attributes of the patient record represent the description of the pathologies of these objects).

First of all, we construct the possibility-based proximity matrix of the objects of CB modeled by the inter-object necessity degrees of proximity by following the steps explained in Section 2. Using the algorithm of the LUS explained in details in [1] to represent the similarity along a linear continuum, and the algorithm of CUS clearly illustrated in [7] to represent the similarity in a closed circular continuum, we get the results presented in Fig. 2(a) for the circular representation and we get the following order for the unidimensional scalling: $x = \{\{O_{15}\} \in P_4, \{O_{17}, O_{16}, O_{18}\} \in P_5, \{O_1, O_2\} \in P_1, \{O_{14}, O_{12}, O_{11}, O_{13}\} \in P_3, \{O_3, O_{10}, O_7, O_8, O_5, O_9, O_6, O_4\} \in P_2\}$, $y = \{-0.71, -0.62, -0.54, -0.45, -0.37, -0.30, -0.20, -0.12, -0.04, 0, 0.13, 0.22, 0.30, 0.38, 0.45, 0.54, 0.63, 0.73\}$, where x represents the ordered objects with their classes, while y represents their corresponding coordinates.

These results show that a strong similarity exists between the objects belonging to the same pathology. In other words, an object belonging to a given pathology is more similar to any other object from the same family than to the other objects belonging to the other pathologies. Thanks to this characteristic, robust retrieval or case diagnostic and reasoning could be achieved here. From the constructed matrix or/and from the obtained categories and coordinates (the sets x and y above) we can study the relationships that exist between the objects belonging to the same class and we can decompose them into other homogeneous groups according to their similarities in order to understand their characteristics or to extract some interesting potential medical rules. Furthermore, we can have an idea about the similarity that exists between the different pathologies. Let us now apply the 3-dimensional multidimensional scaling (3D MDS) algorithm to the possibilistic proximity matrix (see the results in Fig. 2(b)). The same remarks can be deduced concerning the similarity of objects (remark for example that the objects O_{11} and O_{13} are the object the most similar in this base). Note that though we have presented the 33D objects with only 3D space, the representation is still coherent and gives the expected results. The possibilistic proximity matrix can be also represented as additive trees and ultrametric trees using the graph theory techniques, and similar results and conclusion could be obtained (Fig. 2(c) and 2(d)). Note that the objects belonging to the same pathology are attached to the same internal node (note here that the dendrogram provides us with the hierarchical clusters of this set base). Applying the evidential clustering to this data set gives the partitions shown in Fig. 2(f), while applying the additive clustering gives the results shown in Fig. 2(e), where the weights of the membership assigned to each class are given as: $W_{C_1} = 0.1959$, $W_{C_2} = 0.2186$, $W_{C_3} = 0.2802$, $W_{C_4} = 0.3382$, $W_{C_5} = 0.2587$, $W_{C_\Omega} = 0.2297$. The class Ω



(a) Circular Unidimensional Scaling

(b) Multidimensional Scaling

(c) The Additive Tree

(d) The Ultrametric Tree

(e) Additive clustering

(f) Evidential clustering

Fig. 2. The graphical representational models of the possibilistic similarity matrix of *CB*

represents the total ignorance. All these methods confirm the ground truth provided by the expert (the doctor).

As we see here, the representational models that we proposed and applied in this paper capture several images, and in consequence several possible analyses and interpretations of the same dataset from different angles. In these models, dissimilarity is represented as the distances between the objects in the spatial models, as the minimum length of the path that connects any two objects in the graphical models, classetering as a decreasing function of the weights or as basic belief assignments in the additive and the evidential representations, respectively.

5 Discussion and Perspectives

In this paper, a possibilistic approach to estimate the similarity between objects having heterogeneous, imperfect and missing values is proposed. Regarding its structure, this approach is very simple, clear, and close to human reasoning. Concerning its running time, this algorithm is very fast, simple, and appropriate to be adapted in data mining techniques since it is fundamentally based on basic mathematical operators (max, min, etc.) and because it does not require additional pre-processing phases to prepare the data and to estimate the missing values or to deal with the imprecise observations. Unlike the majority methods of similarity, our method doesn't demand conditions and constraints that limit its use as the other prior works [10]-[6].

In this paper our approach has been applied to a digestive database. In the general case, this method could be applied without any modification to any other medical or non-medical database and valuable potential knowledge about the objects could be discovered.

References

1. Solaiman, B., Dahabiah, A., Puentes, J.: Possibilistic pattern recognition in a digestive database for mining imperfect data. WSEAS Transactions on Systems 8(2), 229–240 (2009)
2. Bouchon-Meunier, B.: La logique floue et ses applications. Addison-Wesley Publishing Company, France (1990)
3. Denoeux, T.: Evclus: Evidential clustering of proximity data. IEEE Transaction 34, 95–109 (2004)
4. Le Guillou, C., Cauvin, J.: From endoscopic imaging and knowledge to semantic formal images. In: Lévy, P.P., Le Grand, B., Poulet, F., Soto, M., Darago, L., Toubiana, L., Vibert, J.-F. (eds.) VIEW 2006. LNCS, vol. 4370, pp. 189–201. Springer, Heidelberg (2007)
5. Ruet, M., Rakoto, H., Hermosillo, J.: Integration of experience based decision support in industrial processes. IEEE 7, 1–6 (2002)
6. Bouchon-Meunie, B., Diaz, J., Rifqui, M.: A similarity measure between basic belief assignments. In: IEEE infor. fusion conf. (2006)
7. Meulman, J., Hubert, L., Arabie, P.: Linear and circular unidimensional scaling for symmetric proximity matrices. British J. Math. Statist. Psych. 50, 253–284 (1997)
8. Solaiman, B.: Information fusion concepts. In: From information elements definition to the application of fusion approaches. SPIE proceedings series, vol. 4385 (2001)
9. Lee, M.D., Vickers, D.: Psychological approaches to data visualisation. Defence Science and Technology Organisation Research Report, DSTO-RR-0135 (1998)
10. Zemirline, A.: Définition et fusion de systèmes diagnostic à l'aide d'un processus de fouille de données: Application aux systèmes diagnostics. TELECOM thesis, Université de Rennes (2008)

A New Measure of the Cluster Hypothesis

Mark D. Smucker[1] and James Allan[2]

[1] Department of Management Sciences
University of Waterloo
[2] Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts Amherst

Abstract. We have found that the nearest neighbor (NN) test is an insufficient measure of the cluster hypothesis. The NN test is a local measure of the cluster hypothesis. Designers of new document-to-document similarity measures may incorrectly report effective clustering of relevant documents if they use the NN test alone. Utilizing a measure from network analysis, we present a new, global measure of the cluster hypothesis: normalized mean reciprocal distance. When used together with a local measure, such as the NN test, this new global measure allows researchers to better measure the cluster hypothesis.

Keywords: Cluster hypothesis, nearest neighbor test, relevant document networks, normalized mean reciprocal distance.

1 Introduction

Central to much of information retrieval (IR) is van Rijsbergen's cluster hypothesis: "closely associated documents tend to be relevant to the same requests" [1]. Early measurements of the cluster hypothesis pointed to the potential utility of cluster retrieval [2] and provided explanations of differing IR performance on different document collections [3].

Tombros and van Rijsbergen [4] recast the cluster hypothesis as not solely a property of a document collection but as a concern of a document-to-document similarity measure as applied to a document collection. With this view, we as designers of document-to-document similarity measures want to create similarity measures that *make the cluster hypothesis true* given a document collection and set of search topics.

In Tombros and van Rijsbergen's work, they created query-sensitive similarity measures (QSSMs). These similarity measures aim to focus similarity on the search user's topic. As such, what is considered similar to a given document changes for each search topic. Tombros and van Rijsbergen found that a QSSM has the ability to make the cluster hypothesis more true compared to similarity measures that ignore the user's query.

The current standard for measuring the ability of a similarity measure to make the cluster hypothesis true is Voorhees' nearest neighbor (NN) test [5]. The NN

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 281–288, 2009.

test measures the number of relevant documents found within rank 5 when a similarity measure ranks documents similar to a relevant document, which is effectively the same as measuring the precision at rank 5 (P5, the number of relevant documents found within rank 5 divided by 5).

Voorhees' NN test is notable for several reasons. The NN test says that what matters is whether or not non-relevant documents are ranked before relevant documents when documents are ranked for similarity to a given relevant document. Just because two relevant documents are very similar given a similarity measure does not preclude many non-relevant documents being more similar to the document. Perhaps most important though is that the NN test is comparable across different similarity measures, search topics, and document collections.

The NN test only requires relevant documents to *locally* cluster and cannot distinguish between a set of relevant documents that only locally cluster and a set of relevant documents that are also *globally* clustered. As such, the NN test may falsely report good clustering performance for query-biased[1] similarity measures. To see how this mistake is possible, assume we have a query that has many ($\gg 5$) relevant documents and a P5 of 1. If we query-bias the similarity until the query dominates over the given relevant document, then the rankings for every relevant document will be nearly identical and also have a P5 of 1. Using the NN test, we would declare the clustering performance to be excellent when in fact it could be very poor. The query may be high in precision but low in recall. Thus, all the relevant documents will be close to a few relevant documents but far away from the majority of relevant documents.

For some similarity measures and document collections, the NN test may fail to detect when relevant documents do cluster well. Wilbur and Coffee [6] found that the cluster hypothesis holds for the CISI collection in contrast to the NN test's negative conclusion [5]. Similar to Wilbur and Coffee's work, we utilized an earlier version of our methodology to measure the navigability of the find-similar interaction mechanism [7].

In this paper, we show that the NN test is an insufficient measure of the cluster hypothesis for a set of query-biased similarity measures. While the NN test works well as a measure of local clustering, it fails as a measure of global clustering. We present a new, global measure of the cluster hypothesis. We recommend that the use of the NN test be complemented with our test — each test tells us something different about how well relevant documents cluster.

2 A Global Measure of the Cluster Hypothesis

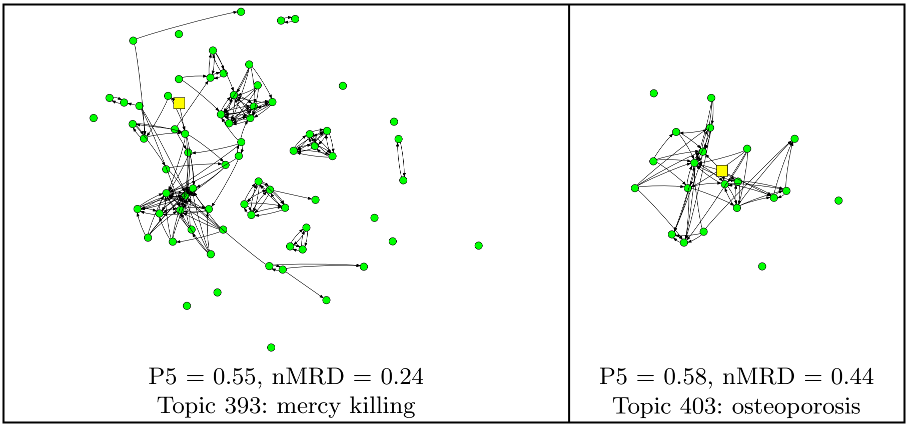
Our new measure of the cluster hypothesis is based on the shortest paths between relevant documents on a directed graph that we construct and call a *relevant document network*. We first describe the construction of the relevant document network and then we present our measure.

[1] We generically refer to similarity measures that bias similarity given the user's query as query-biased. QSSMs are one way to create query-biased similarity measures.

Relevant Document Networks. A document network is a graph where the nodes are documents and the edges of the graph represent some relationship between documents. For our purposes, we construct document networks as fully connected, weighted, directed graphs. An edge from a source document to a target document represents the similarity of the target document to the source when the source is used as a query. Documents and their similarities to each other have long been represented and manipulated in a graph theoretic fashion [1].

Rather than use the similarity measure directly as the edge weight, we set an edge's weight to be equal to the rank of the target document in the ranked results when the source document is used as a query. By weighting links in this manner, we gain the same benefits enjoyed by the NN test described above – notably the ability to directly compare across similarity measures, topics, and document collections. In addition, we exclude the source document from the ranked list of similar documents. We give documents not returned in a ranking an edge weight equal to the number of documents in the collection. Alternatively, one could give such edges a weight of infinity or equivalently not include the link in the graph. For a given source document, no two target documents have the same rank.

Rather than use the whole document network, we delete the non-relevant documents to produce a *relevant document network*. Figure 1 shows two examples of relevant document networks. Since each search topic has a different set of relevant documents, we construct relevant document networks on a per-topic basis.



P5 = 0.55, nMRD = 0.24
Topic 393: mercy killing

P5 = 0.58, nMRD = 0.44
Topic 403: osteoporosis

Fig. 1. Simplified depictions of relevant document networks for TREC topics 393 and 403. Each circular node is a relevant document. A link is drawn from a node to another node if the target node is within the top 5 most similar documents of the source node. The square node in each drawing represents a query likelihood retrieval using the topic's title as a query and is only shown for reference. The actual relevant document networks are fully connected, weighted, directed graphs. In this figure, the document-to-document similarity is "regular" with no weight given to the query/topic, i.e. $\lambda = 0$ (see Section 3).

Normalized Mean Reciprocal Distance. We propose as a new measure of the cluster hypothesis the *global efficiency* measure of Latora and Marchiori [8] applied to a given relevant document network. This measure is based on the shortest path distances between all pairs of vertices in the relevant document network.

This metric computes for each relevant document the normalized, mean reciprocal distance (nMRD) of all other relevant documents. The nMRD of relevant document R_i is calculated as:

$$nMRD(R_i) = \frac{1}{Z(|R|-1)} \sum_{R_j \in R, j \neq i} \frac{1}{D(R_i, R_j)} \tag{1}$$

where R is the topic's set of relevant documents, $|R|$ is the number of relevant documents, $D(R_i, R_j)$ is the shortest path distance from R_i to R_j, and Z is the normalization factor. This metric varies from 0 to 1 with 1 being the best network possible. Because we allow no target documents to have the same rank, the best possible network for a given source document is a complete binary tree and thus:

$$Z = \frac{1}{|R|-1} \sum_{i=1}^{|R|-1} \frac{1}{\lfloor \log_2 i \rfloor + 1} \tag{2}$$

For each topic, we average the nMRD over all the known relevant documents. Finally, for a test collection, we average over all topics to produce a final metric.

Looking again at the example relevant document networks in Figure 1, we see that precision at 5 (P5) reports that both topics 393 and 403 locally cluster relevant documents very well while the global clustering of the two topics is quite different. The normalized mean reciprocal distance (nMRD) reports that the relevant documents are globally much better clustered for topic 403 than for topic 393.

3 Document-to-Document Similarity Measures

In this paper, we use the well known language modeling approach to information retrieval to create a collection of document-to-document similarity measures. In our discussion, we refer to the document to which we are finding similar documents as the *source* document. For a given source document, we will call all other documents in the collection *target* documents.

In all cases, we build a query-biased, multinomial model, M_B, for a given source document and rank the remaining documents in the collection using the Kullback-Leibler divergence:

$$D_{KL}(M_B || M_D) = \sum_w P(w|M_B) \log \frac{P(w|M_B)}{P(w|M_D)} \tag{3}$$

where $0 \log 0 = 0$ and M_D is a smoothed, maximum likelihood estimated (MLE) multinomial model of the target document.

We generate a range of query-biased similarity measures by utilizing two ways to compute a model of the source document, M_S, and then by linearly combining this model with a MLE model of the given topic's query Q to produce a query-biased model M_B:

$$P(w|M_B) = \lambda P(w|Q) + (1 - \lambda)P(w|M_S) \tag{4}$$

where λ varies from 0 to 1 and controls the amount of query-biasing. While the query-biased similarity of Equation 4 is different than Tombros and van Rijsbergen's query sensitive similarity measure (QSSM) [4], which was a measure for vector space retrieval, the above formulation for language modeling retrieval captures the nature of QSSM.

We compute M_S by selecting differing amounts of the source document's text and then letting M_S be the MLE model of this selected text. At one extreme, we select all of the text in a document. When we select the whole document for M_S and set $\lambda = 0$, we produce what we call *regular* similarity, which is the most obvious form of document-to-document similarity and essentially treats a document as a very long query.

We also query-bias the similarity by how we select text from the document for M_S. Our second approach to query-biased similarity aims to capture the context of the query directly by only including the document text near query term occurrences. This "window" approach creates a MLE model of the source document text that consists of all words within a certain distance W of all query terms in the document. In effect, we place windows of size $2W+1$ centered over all query term occurrences in the document. For example, when $W = 5$, the window includes the 5 preceding words, the query term, and the 5 words following the query term. When selecting text in this fashion, if a document does not contain any query terms, the whole document is used.

Besides testing the "window" version of query-biased similarity alone by keeping $\lambda = 0$, we also take the query-biased model of the document that the "window" approach produces and mix this model with the MLE model of the query.

In summary, we have two ways of query-biasing the similarity. The first way mixes the query with a model of the source document M_S. The second way query-biases M_S by letting M_S be a MLE model of the text falling inside windows placed over query term occurrences in the source document. By comparing these two versions of query-biased similarity, we can see if the context captured by the windows holds an advantage over simply mixing the query with the whole document.

4 Experiments

We compared the NN test (P5) and the normalized mean reciprocal distance (nMRD) as measures of the cluster hypothesis on 150 TREC topics and 36 variations of document-to-document similarity.

To produce the 36 types of similarity, we took Equation 4 and investigated λ with values of 0, 0.1, 0.25, 0.5, 0.75, and 0.9. Besides utilizing the whole

P5

	Window Size W for M_S					
λ	All	15	10	5	2	1
0.90	0.45	0.45	0.45	0.45	0.45	0.45
0.75	0.46	0.47	0.47	0.47	0.48	0.48
0.50	0.51	0.52	0.52	0.53	0.52	0.50
0.25	0.56	**0.58**	0.57	0.57	0.54	0.50
0.10	0.51	0.53	0.53	0.53	0.50	0.47
0.00	0.32	0.40	0.42	0.44	0.44	0.42

nMRD

	Window Size W for M_S					
λ	All	15	10	5	2	1
0.90	0.15	0.15	0.16	0.16	0.16	0.16
0.75	0.17	0.17	0.17	0.17	0.18	0.18
0.50	0.20	0.21	0.21	0.22	0.22	0.21
0.25	0.27	0.29	0.29	0.29	0.28	0.25
0.10	0.30	**0.32**	**0.32**	**0.32**	0.30	0.27
0.00	0.20	0.27	0.28	0.29	0.29	0.26

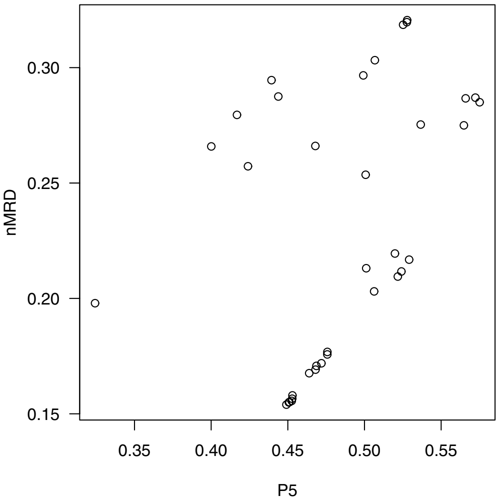


Fig. 2. Precision at 5 (P5) and normalized mean reciprocal distance (nMRD) measures of the cluster hypothesis for 36 variations of document-to-document similarity. The parameters λ and W refer to Equation 4 with "All" meaning the whole document is used to compute the source document model M_S. Scores without a statistically significant difference from the best scores are in **bold**. We measured statistical significance using the paired Student's t-test ($p < 0.05$). The plot on the right shows nMRD vs. P5.

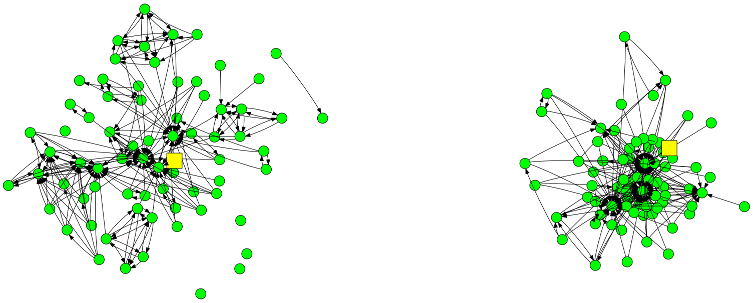
document to compute M_S in Equation 4, we also investigated window sizes W of 1, 2, 5, 10, and 15 words. The six settings of λ and six different window sizes for computing M_S produce the 36 similarity measures.

For our queries, we used the title field of TREC topics 301-450, which are the topics for TREC 6, 7, and 8. The document collection consists of TREC volumes 4 and 5 minus the Congressional Record. We smoothed the M_D of Equation 3 using Dirichlet prior smoothing with its parameter set to 1500. We truncated the query model M_B of Equation 4 to its 50 most probable terms. We stemmed using the Krovetz stemmer and used an in-house list of 418 stop words. We used the Lemur toolkit for our experiments.

5 Results and Discussion

Figure 2 shows the results for our experiments. While not shown here, we have found little difference relative to nMRD between P5 and the use of P10, P20, and average precision as local measures [9].

While generally a higher score for the NN test (P5) implies a higher score for the global measure (nMRD), there are numerous runs where the document-to-document similarity measure produced results with high P5 but with low nMRD. For example, the P5 measure has a value of 0.53 for the runs with a



$\lambda = 0.1$, P5 = 0.63, nMRD = 0.33

$\lambda = 0.25$, P5 = 0.57, nMRD = 0.19

Fig. 3. Simplified depictions of relevant document networks for topic 393 with $\lambda = \{0.1, 0.25\}$ (see Equation 4). Figure 1, shows topic 393 with $\lambda = 0$. Both Figure 1 and this figure compute M_S by using all the text of the source document.

window size $W = 5$ words and the λ values of 0.1 and 0.5, but when $\lambda = 0.1$, nMRD = 0.32 and when $\lambda = 0.5$, nMRD drops to 0.22. The NN test as a local measure of the cluster hypothesis is unable to measure the global clustering of relevant documents. If the NN test is used alone, it is possible to develop similarity measures that falsely appear to cluster documents well.

Nevertheless, to obtain a more complete view of the cluster hypothesis, both a global and local measure are needed. There are many similarity measures that produced relatively high nMRD scores between 0.25 and 0.3 while at the same time resulting in P5 scores ranging from 0.40 to 0.58. The nMRD measure is unable to detect local clustering of relevant documents.

Setting $\lambda = 0.1$ produced the best nMRD scores for all context sizes. Using a reduced context of 5, 10, or 15 words produced slightly better results than using the whole document (nMRD of 0.32 versus 0.30). For this document collection, it appears that there is some value to the window form of query-biased similarity although the majority of the benefit seems to come from giving the original query enough, but not too much weight.

The lower nMRD scores for the high values of λ are likely the result of a lack of diversity in the similarity lists across documents. Giving the query too much weight produces a ranking of similar documents that is more or less the same for all documents. From each document it becomes easy to traverse the relevant document network to a few relevant documents, but once at these documents, there is no easy way to travel to other relevant documents.

For topics such as topic 393 (Figures 1 & 3), we see that with a query-biased similarity, many of the outlying documents now have 2 or 3 of the query-similar documents as top ranked similar documents. These same outlying documents though have failed to gain connections to each other. Here it seems that query-biased similarity may be making the cluster hypothesis more true only by moving a few relevant documents closer to all relevant documents but not by helping all of the relevant documents get closer to each other.

While query-biased similarity has made the cluster hypothesis more true, the resulting connections between relevant documents are likely not robust to deletion of key, query-similar documents. If the query-similar documents did not exist in the collection, query-biased similarity might have had a less dramatic effect on the clustering of relevant documents. In addition to their global efficiency measure, Latora and Marchiori [8] have a local measure of efficiency that concerns itself with this question of robustness. In future work, we'd like to examine Latora and Marchiori's local efficiency measure and investigate to what extent similarity measures produce fault tolerant relevant document networks.

6 Conclusion

In this paper we presented a new measure the cluster hypothesis: normalized mean reciprocal distance (nMRD). This new measure is based on the shortest paths between documents on a relevant document network. In contrast to the NN test, which is a local measure of clustering, nMRD is a global measure of the cluster hypothesis. We examined 36 variations of document-to-document similarity and showed that the NN test is not a sufficient measure of the cluster hypothesis. Different similarity measures can score well on the NN test but have very different scores on the global measure, nMRD. To better determine the ability of similarity measures to make the cluster hypothesis true, both a global and local measure should be used.

Acknowledgments. This work was supported in part by the Center for Intelligent Information Retrieval, and in part by the University of Waterloo. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors' and do not necessarily reflect those of the sponsors.

References

1. van Rijsbergen, C.J.: Information Retrieval, 2nd edn., Butterworths (1979)
2. Jardine, N., van Rijsbergen, C.J.: The use of hierarchic clustering in information retrieval. Information Storage and Retrieval 7(5), 217–240 (1971)
3. van Rijsbergen, C.J., Sparck Jones, K.: A test for the separation of relevant and non-relevant documents in experimental retrieval collections. Journal of Documentation 29, 251–257 (1973)
4. Tombros, A., van Rijsbergen, C.J.: Query-sensitive similarity measures for the calculation of interdocument relationships. In: CIKM 2001, pp. 17–24 (2001)
5. Voorhees, E.M.: The cluster hypothesis revisited. In: SIGIR 1985, pp. 188–196 (1985)
6. Wilbur, W.J., Coffee, L.: The effectiveness of document neighboring in search enhancement. IPM 30(2), 253–266 (1994)
7. Smucker, M.D., Allan, J.: Measuring the navigability of document networks. In: SIGIR 2007 Web Information-Seeking and Interaction Workshop (2007)
8. Latora, V., Marchiori, M.: Efficient behavior of small-world networks. Physical Review Letters 87(19) (October 2001)
9. Smucker, M.D.: Evaluation of Find-Similar with Simulation and Network Analysis. PhD thesis, University of Massachusetts Amherst (2008)

Explaining User Performance in Information Retrieval: Challenges to IR Evaluation

Kalervo Järvelin

University of Tampere, Finland
kalervo.jarvelin@uta.fi

Abstract. The paper makes three points of significance for IR research: (1) The Cranfield paradigm of IR evaluation seems to lose power when one looks at human instead of system performance. (2) Searchers using IR systems in real-life use rather short queries, which individually often have poor performance. However, when used in sessions, they may be surprisingly effective. The searcher's strategies have not been sufficiently described and cannot therefore be properly understood, supported nor evaluated. (3) Searchers in real-life seek to optimize the entire information access process, not just result quality. Evaluation of output alone is insufficient to explain searcher behavior.

1 Introduction

The dominant view on IR theory boils down to formal models of information retrieval (IR). These models are abstract specifications for the search engines to work – quite different from empirical theories, which one confirms or refutes in a lab or in the real world. If the mathematics make sense and the implementation is faithful to the model, the engine will work. Search engine effectiveness, on the other hand, cannot be tested within the formal model alone; for that one needs some experimental instrumentation and evaluation. We ask in this paper, how much and what kind of theoretical understanding the IR community has regarding IR effectiveness. The retrieval models do not cover these aspects – or at best, make strong implicit assumptions about it.

The goals of a research area may be classified as (a) theoretical understanding, (b) empirical description, prediction and explanation, and (c) technology development in the domain of interest. Much of research in IR is driven by a technological interest of developing tools for information access. However, technological interest becomes blind if not nurtured by the other goals. [6]

Reflecting this, the motivation for the present paper is that the ultimate goal of information retrieval is to support humans to better access information in order to better carry out their task. How well does IR effectiveness, measured at the output of search engines, reflect this? If IR effectiveness does not directly translate to better human information access, we risk turning means to ends with unfortunate consequences.

In the present paper, we make three points: (1) The Cranfield style of IR evaluation seems to lose power when one looks at human instead of system performance. (2) Searchers using IR systems in real-life use rather short queries, which individually often have poor performance. However, when used in sessions, they may be

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 289–296, 2009.

surprisingly effective. The searcher's strategies have not been sufficiently described and cannot therefore be properly understood, supported nor evaluated. (3) Searchers in real-life seek to optimize the entire search process, not just result quality. Evaluation of output alone is insufficient to explain searcher behavior.

In Section 2, we review some past research on (non-formal) IR theory and introduce some concepts for discussing research approaches or paradigms. Section 3 discusses the limitations of the Cranfield approach in the light of recent empirical evidence. Section 4 takes a look at real-life IR based on sessions of short queries and argues that the Cranfield approach can be extended in this direction. Section 5 proposes a more holistic approach to IR evaluation based on searcher costs and efforts as well as output quality. Section 6 contains conclusions.

2 Past Analyses of IR Research

There are several introductions to approaches in IR research. For example, [6] reviewed three major approaches: systems-oriented IR, user-oriented IR and cognitive IR approaches. Järvelin [9] discussed the models and theories of systems-oriented and cognitive IR approaches. There is a dominant model for systems-oriented IR research, the Cranfield evaluation approach based on test collections. The other two major approaches do not have such dominant models.

Saracevic discussed critically evaluation in IR research and called for the integration of user-oriented and system-oriented IR research [13]. He criticized the sole use of relevance-based measures in evaluation and called for proper measures at the levels of users and uses, markets and products, and social impacts.

Ellis [3] questioned the applicability of the results of the Cranfield approach to operational systems due to validity problems in performance evaluation. He pointed out that the approach abstracts a mechanical component out of human interaction with texts at the cost of not being able to handle problems at the searcher level.

3 The Focus and Limits of Cranfield Style of IR Evaluation

3.1 The Cranfield Framework

Figure 1, center and left, represents the essence of Cranfield style of IR evaluation – the core components of experimental designs. In the unshaded center area there are documents/collections, requests/queries, and results; and core processes of representation and matching with some feedback. Their interaction is however laborious to study in real-life. We therefore want to move the components into a lab, and incorporate the shaded area, the necessary lab instrumentation. Using the instrumentation: standard collections, search requests, relevance assessments, and evaluation procedures and metrics, enables us to effectively evaluate IR techniques and compare the results. Prototypically, the context and the user are excluded in experiments.

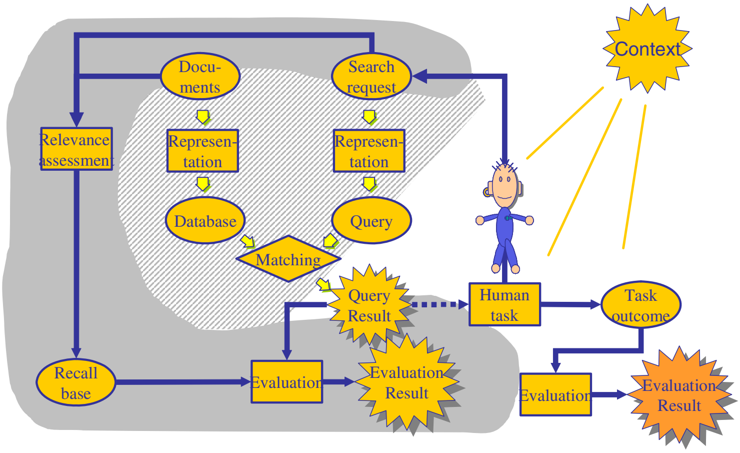


Fig. 1. Evaluation by human performance (extended from [9])

3.2 Limitations of the Cranfield Approach: WYDSIWYDU

Cranfield style of IR research develops techniques for finding relevant documents. The quality of the techniques is usually measured through recall and precision of the output, or metrics based on these (e.g. MAP). The studies seek to explain the variation of output quality. The independent variables are the use or non-use of various IR techniques and the controlled ones the test collections, topics, assessments. [9]

The Cranfield framework has indispensable benefits that have lead to great progress. Standardization of experimental designs facilitates comparison of findings. One should not, however, be blinded by this success. A study designed within the Cranfield framework cannot claim anything about external variables: WYDSIWYDU – *what you don't see is what you don't understand.* These cover the tasks and searchers supported, relevance assessment, interface functionalities, and the actual search processes. There is mounting evidence that we may not be able to improve human performance by further improving traditional retrieval effectiveness.

3.3 Human Performance

Several recent studies have suggested that using a better search system may not always lead to improvements in task outcomes. Note here that we are stepping out of the lab, measuring something that essentially lies outside – the right side of Figure 1.

Allan and colleagues [1] studied searcher productivity in a passage-based question answering task. User performance improved significantly given a system performance improvement (from bpref of 50) whereafter system performance improvements did not yield a significant user performance change.

Turpin and Scholer [15] studied user performance on simple web search tasks, considering the time that a user takes to find a relevant document, and the number of relevant documents that a user can find within 5 minutes. This was studied across search systems operating at MAP in the range of 0.55 to 0.95. Results indicated that

MAP level has no significant relationship with the time taken to find the first answer, while there was a weak relationship with the recall-oriented task.

Smith and Kantor [14] also explored the relation of system performance to search behavior. Their test subjects each completed several searches using either a standard system or a degraded system. Searchers using degraded systems were *as successful* as those using the standard one, regarding the quality of documents found and the time taken to achieve this. However, searchers using degraded systems *altered their behavior*, making significantly more queries and examining shorter lists.

Huuskonen and Vakkari [5] studied the connection of searching features to task outcome and found and found very few and vague connections between work task result quality and the system/searching variables.

These studies suggest that if one extends the Cranfield framework toward the human tasks, it loses strength. The main dependent variable, traditional IR effectiveness, is only weakly related to human task performance. Consequently, typical IR variables – IR techniques – do not explain the variation in the human task. Further, if the effect of the query result, measured through recall-precision metrics, is only weakly connected to human task performance, then:

- no experimentation with retrieval models will change the situation;
- no variation of evaluation metrics will change this if the metrics remain traditional;
- we need, in addition to result metrics, metrics for the process and the outcome.

4 Interaction in Sessions

Much IR research is based on batch mode experiments where a topic is automatically converted to a single multi-word query, which is then run against the database using some search engine. In real-life, searchers use very short queries but may try out multiple queries in a session. They also behave individually during search sessions. Their information needs may initially be muddled and change during the search process; they may learn as the session progresses, or switch focus. The initial query formulation may not be optimal and the searchers may need to try out different wordings. [10]

Real-life searchers often prefer short queries and avoid excessive browsing. Jansen and colleagues [8] analyzed transaction logs of thousands of queries posed to a Web search engine. The average query length was 2.21 keywords. Less than 4 % of the queries had more than 6 terms. They also observed that most users did not access results past the first page. Therefore real life sessions often consist of sequences of short queries. The data in Table 1 reflect these findings.

The data for Table 1 come from an empirical, interactive IR study [10]. Thirty domain experts each completed the same four realistic search tasks A – D simulating a need for specific information required to make a decision in a short time frame. Each task formed a session. The data show great variability between the tasks along various variables. On average, there were 2.5 queries per session and 2.4 unique keys per session, and each query had two keys and 0.9 filters (a geographic, document type or other condition). Only 10 among the 60 sessions employed four or more unique search keys. These searchers were precision-oriented, i.e., they quit searching soon after finding one or a few relevant documents.

Table 1. Real-life session statistics based on 15 sessions for Tasks A-D (N=60) sessions

Variable	A	B	C	D	Tot
Tot # queries per task	25	59	28	40	152
Avg queries in session	1.7	3.9	1.9	2.7	2.5
Avg # keys per session	1.5	3.9	1.9	2.2	2.4
Avg # keys per query	1.4	2.4	1.8	2.0	2.0
Avg # filters per query	1.2	1.1	0.8	0.7	0.9
S1 frequency	11	3	4	3	21
S2 frequency	2	4	3	4	13
S3 frequency	4	13	11	10	38
S1-S3 frequency sum	17	20	18	17	72

The four bottoms lines report the *frequency of the query strategies* discussed below. The strategies S1, S2, and S3 were identified in Table 1 session data through a secondary analysis [11]. Strategy S1 consisted of individual words used alone as queries. If the first word was unsuccessful, another was tried instead. S1 was employed 21 times in the 60 sessions of Table 1. Strategy S2 is based on incremental query extension: a searcher starts with a one word query. If it is not successful, (s)he extends the query by another word, by a third word, etc. S2 was employed in 13 times of the 60 sessions of Table 1. Strategy S3 is based on three word queries where the first two are fixed and the third one varied. S3 was employed in 38 of the 60 sessions. The total number of identified strategies (72) exceeds the number of sessions (60) because more than one strategy was employed in some sessions. For completeness, strategy S4 is defined as a full multiword query based on a test topic – no-one used it in the empirical data.

Keskustalo and colleagues [11] used the strategies S1-S4 in a simulation experiment based on TREC 7 and 8 test collections (528155 documents) and 41 topics for which graded relevance assessments were available. The retrieval system was *Lemur*. The authors used real test persons to suggest keywords for queries of various lengths. These were then used to construct simulated sessions following the Strategies S1-S4 with an interest in finding whether sessions based on simple queries and Strategies S1-S3 are competitive with verbose individual queries using Strategy S4. Sessions of five queries were used for strategies S1-S3 and the search task was to find one highly relevant document, which is a frequently used task in interactive IR experiments. For each of 5 queries in S1-S3, only the first result page was examined. For S4, the top-50 was examined in lots of ten results for compatibility.

Taken individually, the queries in sessions of Strategies S1-S3 often had poor effectiveness (e.g. measured by MAP). However, session effectiveness of S1-S3 was considerably higher (Table 2). The average page of success tells which page, on the average, contained the first highly relevant document. Based on stringent relevance criteria, S1 is 20-34 percent units weaker than strategies S2-S4 in its success rate. Strategies S2-S3 are only 8-13 % units below S4. In all cases, the first highly relevant document is found, on average, by the second attempt (second page for S4). According to Friedman's test the differences between the strategies are highly significant ($p < 0.001$). In pairwise tests, S3 is not significantly different from S4 while S2-S4 are all significantly better than S1 ($p<0.01$).

Table 2. Effectiveness of session strategies S1-S4 for 41 topics as average page of success

Variable	S1	S2	S3	S4
Avg successful page	1,82	1,73	1,52	1,42
Success rate	23	33	31	36
Success %	60,5	86,8	81,6	94,7

This study shows that sessions based on individually ineffective queries may be surprisingly effective. The findings motivate the observed real-life user behavior, which real users must have learned through experience. As few very simple attempts often lead to good enough results, there is no incentive to pay more effort.

5 Toward a Holistic View on IR

We believe that research on IR interaction is currently too exclusively focused on the quality of retrieval results. Early papers on IR evaluation had a comprehensive approach: Cleverdon and colleagues [2] and Salton [12] identified, among others, presentation issues and intellectual and physical user effort as important factors in IR evaluation, along with recall and precision as performance measures. Usability studies also have a comprehensive approach to costs and benefits of systems assessed [7].

Hersh [4] pointed out that the potential impact of an interactive IR system is determined in part by situational relevance, which is affected, among others, by the user's time pressure. Therefore only documents retrieved in the top ranks of results may be of interest. Järvelin and colleagues [10] extended the Discounted Cumulated Gain metric into a session-based evaluation metric (sDCG), which handles multiple query sessions and takes the searcher's effort (both scanning and query modification costs) indirectly into account through discounting factors.

While costs and benefits of interactive IR systems have been discussed in the literature, the same does not hold for current IR evaluation, which seems to focus on retrieval result quality and neglect searcher efforts. In interactive settings both (expected) costs and benefits affect searcher behavior and evaluation becomes biased if only result quality is considered. To avoid this problem, a cost/benefit model for interactive IR sessions is needed. It should incorporate at least the following cost/benefit factors in a typical search interface:

- Search key generation cost (K): the cost of producing each search key.
- Query execution cost (Q): the cost of giving a search and waiting for the result.
- Result scan cost (S): The cost of scanning each item in the result.
- Next page access cost (N): the cost of accessing the next results page for scanning.
- Relevant document gain (G): the benefit of identifying a relevant document.

A rough cost/benefit model assumes all the above costs linear per respective numbers of units, in the same value range (e.g. seconds), and additive. When this is made commensurate with the relevant document gain, e.g. by a conversion factor between costs and gains, one may use the following function *SessionCBA* for evaluation:

$$\mathit{SessionCBA}(\mathrm{K,Q,S,N,G}) = \alpha\mathrm{K} + \beta\mathrm{Q} + \delta\mathrm{S} + \gamma\mathrm{N} + \theta\mathrm{G} \qquad (1)$$

where α, β, δ, γ, and θ are constant unit costs/benefits of the above variables K, Q, S, N, and G. Note that traditional IR evaluation assumes $\alpha = \beta = \delta = \gamma = 0$ and $\theta > 0$, thus making *SessionCBA*(K,Q,S,N,G) = θG and focusing on benefits at any cost. This can hardly be used to explain searcher behavior.

As an example, consider the case in Section 4 by [11]. Table 3 gives the expected number of search keys K, queries Q, scanned documents S, fetched next pages N, and found relevant documents G (one in each case) for each strategy.

Table 3. Cost-benefit features of Strategies S1-S4

Strategy	K	Q	S	N	G
S1	8.6	3.5	35	0.0	1
S2	4.3	2.3	23	0.0	1
S3	7.3	2.4	24	0.0	1
S4	16.9	1.0	5	0.0	1

The cost-benefit features of Strategies S1-S3 are calculated based on the success statistics of 1st – 5th queries and on the assumption that, if none of them is successful (see Table 2), that the searcher would launch one more query represented by S4 containing 16.9 search keys. If the action would be just giving up without an answer after five unsuccessful attempts, the K column would have values 3.9, 2.6, 5.1, and 16.9 keys. This may happen, if target information is not very valuable.

Because we do not know the unit costs of K, Q, S, and N, we cannot directly identify an optimal strategy. One may still observe that if entering query words is costly and scanning the result cheap, S1-S3 are competitive, whereas in the opposite case S4 wins.

6 Discussion and Conclusion

Theoretical growth in a research area may incur from theory expansion (e.g., through new concepts), greater analytical power (through model building), improved empirical support, and proliferation of new hypotheses within the theory [16].

Section 3 discussed the Cranfield IR evaluation framework and its limitations in the light human task performance. Based on several critical studies we found evidence suggesting that the Cranfield style IR evaluation framework is weakly connected to human task performance. In the effort of making experiments controllable, the Cranfield approach may have crystallized a study design that weakly relates to the activity supported. No experimentation with IR techniques or traditional evaluation metrics will change the situation. To explain user performance theory expansion is necessary.

We then discussed interaction in real-life IR sessions and discussed three idealized real-life, session-based, retrieval strategies S1-S3 as alternatives to a long test query S4. A simulated interactive retrieval experiment showed that sessions using individually ineffective queries may be surprisingly effective. The findings motivate the observed real-life user behavior, which real users must have learned through experience with IR systems. This suggests that greater analytical power is needed for understanding user behavior. Section 5 exemplifies that this can be achieved by more holistic modeling of both session costs and benefits for better empirical support.

In conclusion, there are risks in focusing wholly on IR tools without analyzing their real use contexts. One cannot understand their use, nor design them properly, without understanding at least minimally the information environment of their users.

Acknowledgement

This research was funded by the Academy of Finland grants #120996 and #124131.

References

1. Allan, J., Carterette, B., Lewis, J.: When will information retrieval be “good enough”? In: Proc. ACM SIGIR 2005, pp. 433–440. ACM Press, New York (2005)
2. Cleverdon, C., Mills, L., Keen, M.: Factors determining the performance of indexing systems, vol. 1 - design. Aslib Cranfield Research Project, Cranfield (1966)
3. Ellis, D.: Progress and problems in information retrieval. Library Assoc., London (1996)
4. Hersh, W.: Relevance and Retrieval Evaluation: Perspectives from Medicine. J. Amer. Soc. Inform. Sci. 45, 201–206 (1994)
5. Huuskonen, S., Vakkari, P.: Students’ search process and outcome in Medline in writing an essay for a class on evidence based medicine. J. Documentat. 64, 287–303 (2008)
6. Ingwersen, P., Järvelin, K.: The turn: Integration of information seeking and retrieval in context. Springer, Heidelberg (2005)
7. ISO Ergonomic requirements for office work with visual display terminals (VDTs), Part 11: Guidance on usability. ISO 9241-11:1998(E) (1998)
8. Jansen, B.J., Spink, A., Saracevic, T.: Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web. In: Inform. Proc. Manag., vol. 36, pp. 207–227 (2000)
9. Järvelin, K.: An Analysis of Two Approaches in Information Retrieval: From Frameworks to Study Designs. J. Amer. Soc. Inform. Sci. 58, 971–986 (2007)
10. Järvelin, K., Price, S.L., Delcambre, L.M.L., Nielsen, M.L.: Discounted Cumulated Gain Based Evaluation of Multiple-Query IR Sessions. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 4–15. Springer, Heidelberg (2008)
11. Keskustalo, H., Järvelin, K., Pirkola, A., Sharma, T., Lykke Nielsen, M.: Test Collection-Based IR Evaluation Needs Extension Toward Sessions - A Case Study of Extremely Short Queries. In: Proc. AIRS 2009. LNCS, Springer, Heidelberg (to appear, 2009)
12. Salton, G.: Evaluation Problems in Interactive Information Retrieval. Inform. Stor. Retr. 6, 29–44 (1970)
13. Saracevic, T.: User lost: reflections on the past, future, and limits of information science. ACM SIGIR Forum 31(2), 16–27 (1997)
14. Smith, C.L., Kantor, P.B.: User Adaptation: Good Results from Poor Systems. In: Proc. ACM SIGIR 2008, pp. 147–154. ACM Press, New York (2008)
15. Turpin, A., Scholer, F.: User performance versus precision measures for simple search tasks. In: Proc. ACM SIGIR 2006, pp. 11–18. ACM Press, New York (2006)
16. Wagner, D., Berger, J., Zeldith, M.: A working strategy for constructing theories. In: Ritzer, G. (ed.) Metatheorizing, pp. 107–123. Sage, Tousand Oaks (1992)

A Four-Factor User Interaction Model for Content-Based Image Retrieval

Haiming Liu[1], Victoria Uren[1,⋆], Dawei Song[2], and Stefan Rüger[1]

[1] Knowledge Media Institute, The Open University, Milton Keynes, UK
[2] School of Computing, The Robert Gordon University, Aberdeen, UK
{h.liu,s.rueger}@open.ac.uk, v.uren@dcs.shef.ac.uk, d.song@rgu.ac.uk

Abstract. In order to bridge the "Semantic gap", a number of relevance feedback (RF) mechanisms have been applied to content-based image retrieval (CBIR). However current RF techniques in most existing CBIR systems still lack satisfactory user interaction although some work has been done to improve the interaction as well as the search accuracy. In this paper, we propose a four-factor user interaction model and investigate its effects on CBIR by an empirical evaluation. Whilst the model was developed for our research purposes, we believe the model could be adapted to any content-based search system.

Keywords: User interaction, Relevance feedback, Content-based image retrieval.

1 Introduction

Content-based image retrieval (CBIR) has been researched for decades, but it is not widely applied online. In our view, one of the reasons for this is that CBIR is normally performed by computing the dissimilarity between objects and queries based on their multidimensional feature vectors in content feature spaces, for example, colour, texture and structure features. There is a well known gap, called the "semantic gap", between the low-level feature of an image and its high-level meaning to users.

To help bridge this semantic gap, relevance feedback (RF) has been introduced into CBIR systems, which aims to bring users into the search loop. Existing research on RF [10] suggests that bringing users into the loop can help bridge the semantic gap and may also improve the retrieval accuracy. However, most existing RF techniques are highly system-centric. They focus more on improving search accuracy than the interaction between the system and users.

Therefore in an effort to develop more human-centric and user-oriented systems, Spink, et al. proposed a three-dimensional spatial model to support user interactive search for text retrieval [8]. The model emphasizes that partial relevance is as important as binary relevance/irrelevance, and indeed it can be more important for inexperienced users.

[⋆] Present address: Department of Computer Science, Regent Court, 211 Portobello, University of Sheffield, Sheffield, S1 4DP United Kingdom.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 297–304, 2009.

Other existing research has been focused more on a single dimension, such as time. For example, Campbell in [1] proposed the Ostensive Model (OM) that indicates the degree of relevance relative to when a user selected the evidence from the results set. Later, Urban, et al. applied the so called increasing profile to CBIR [9]. Their preliminary study showed that the system based on the OM was preferred by users over traditional CBIR search engines.

Ruthven, et al. [6] adapted two dimensions from the Spink, et al model combined with OM in their study. Their experimental results showed that combining partial and time relevance did help the interaction between the user and the system.

Based on the related work, we are motivated to investigate what the outcome would be were we to combine the three-dimensional spatial model with the OM together and, further, to add another factor - frequency - to the combination. Therefore, in this paper, we propose an adaptive four-factor user interaction model (FFUIM) including relevance region, relevance level, time and frequency.

We will investigate the different interaction settings of the FFUIM, through simulated evaluations on a large image collection. The evaluation results provide initial evidence and insights into which interaction settings are likely to deliver the best search accuracy and lead to better user search experience.

2 User Interaction Models

In this section, we review a number of existing UI models and describe how our FFUIM harnesses their advantages, whilst addressing some of their limitations.

2.1 Three-Dimensional Spatial Model

In order to improve the interaction between the users and the system, Spink, et al. proposed a three-dimensional spatial model of levels of relevance, regions of relevance and time of relevance to text retrieval [8]. Firstly, they applied Saracevic's five levels of relevance [7] as the way to indicate why the feedback is relevant, which confers a qualitative difference between levels. Secondly, the regions of relevance indicate the degree of users' relevance judgements to a feedback. The four regions are relevant, partially relevant, partially not relevant and not relevant. The third dimension is time of relevance, which is measured in formats such as information seeking stage and successive searches. We consider the model as a useful foundation from which to develop further user interaction models and techniques for CBIR.

2.2 Ostensive Model

Other research has tended to focus more on a single dimension, such as time. For example, Campbell in [1] proposed the Ostensive Model (OM) that indicates the degree of relevance relative to when a user selected the evidence from the results set. OM includes four ostensive relevance profiles: decreasing, increasing, flat and current profiles, respectively. With the increasing profile the latest RF is deemed most important, whereas with the decreasing profile it is the earliest RF that is regarded as the most important. With the flat profile all RF is given equal importance, regardless of when the feedback was provided. Finally, the current profile gives the latest RF the highest weight and earlier RF is ignored. Campbell found that for text retrieval the increasing, flat and current profile showed overall better accuracy than the decreasing model, and the increasing profile was the most robust [1].

In [9] Urban et al. adapted the OM from text retrieval for CBIR to help overcome interaction problems between users and CBIR systems. In that study only the increasing profile was applied. The results indicated that, whilst users found the OM easy to use, they found it difficult to control the RF process without greater interaction. Furthermore, the traditional OM accepted only positive RF, whereas in reality users wish to refine their searches by providing both negative and positive RF. Indeed, some research [2,5,4] has shown that including negative examples into the RF can actually help improve the image retrieval accuracy.

2.3 Partial and Ostensive Evidence

Ruthven, et al. [6] adapted two dimensions from Spink, et al. model, namely: regions of relevance and time, for ranking query expansion terms in text retrieval. The region of relevance in their study is called partial evidence, which is a range of relevance level from one to ten. In addition, they applied the OM to the time dimension, which is called ostensive evidence. The ostensive evidence is measured by iterations of feedback. Their study shows that combining RF techniques with the user interaction factors is preferred by users over RF techniques alone. It will be interesting to see how the combined model performs in our CBIR system.

3 A Four-Factor User Interaction Model for CBIR

Based on these interesting studies, we developed a new model named ‘four-factor user interaction model (FFUIM)’, which combines the three-dimensional spatial model with the OM and, further, to add another factor - frequency - to the combination. The FFUIM includes: relevance region, relevance level, time and frequency. We introduce the four factors in following sections.

3.1 Relevance Region

Instead of Spink, et al. four regions of relevance, the relevance region here comprises two parts: relevant (positive) evidence and non-relevant (negative) evidence. Both relevance regions contains a range of relevance levels.

3.2 Relevance Level

The relevance level here indicates how relevant/non-relevant the evidence is on the related relevance region, which implies a quantitative difference, and differs from Saracevic's definition in Spink, et al. This factor is measured by a range of relevance level (integer 1-20) indicated by users. The distance function with the relevance level factor is given by

$$D_{ij} = d_{ij}/W_p, \tag{1}$$

where $D_{ij}(i = 1, 2, \ldots, m; j = 1, 2, \ldots, n)$ is the final distance between a query image i with an object image j; d_{ij} is the original distance between the query image i and an object image j; W_p is the partial weight, $W_p = r$ for the positive examples, and $W_p = \frac{1}{r}$ for the negative examples (r is the level of the relevance provided by the user between 1 and 20 integer)[1] [2].

3.3 Time

We adapted the OM to the time factor to indicate the degree of relevance relative to when the evidence was selected. In this study, we have taken the OM a step further. In addition to using the increasing profile, we have also tested the flat profile, current profile and the decreasing profile. For our study, the increasing / decreasing profile means ostensive relevance weights for positive / negative examples increase / decrease respectively with further search iterations. The fundamental difference between our studies and Urban et al. is that we have applied these ostensive relevance weights to both the positive and negative feedback, and applied the weight to more than one image in every query. We propose the following distance function with ostensive weight:

$$D_{ij} = d_{ij}/W_o, \tag{2}$$

where W_o, the ostensive weight, can be different depending on the profile. $W_o = s$ for the positive examples, and $W_o = \frac{1}{s}$ for the negative examples (for the increasing profile, s is iterations of feedback; for the decreasing profile, s is iterations of feedback in the contrary order; for the flat profile, s is 1; for the current profile, s is 1 to current iteration, but 0 to previous iterations)[3].

3.4 Frequency

While we were investigating the combined models, we found that the same images can be used as positive/negative examples in different RF iterations. Thus, we wonder: can the number of times an image appears (frequency) across all the

[1] D_{ij} depending on positive d_{ij}/x and negative examples $d_{ij}/(1/x)$, but the later simplifies to $d_{ij} \times x$, here the x can be r,s,t. Therefore the distance become smaller the higher the positive weight and larger the higher the negative weight.

[2] Note that we have tested a number of other weighting functions for W_y (y can be o,p,f), e.g., $W_y = x$, $W_y = 2^x$ and $W_y = \ln(x)$ (x can be r,s,t) for positive examples, but there was no significant difference in performance (MAP). Here we use the linear setting for simplicity.

[3] Please see more detail in footnote 1 and 2.

iteration contribute to the model? To answer this question, we propose a new factor - frequency, which captures the number of appearances of an image in the user selected evidence both for positive and negative evidence separately. The distance function with frequency is given by

$$D_{ij} = d_{ij}/W_f, \tag{3}$$

where W_f, the frequency weight, is how often an image has been chosen as a relevant or non-relevant example: $W_f = t$ for the positive examples, and $W_f = \frac{1}{t}$ for the negative examples (t is the number of times the image was chosen as a feedback)[4].

4 Empirical Evaluation

Our empirical experiments aim to find possible interaction settings of the FFUIM that improve the search accuracy in comparison with a CBIR system without any interaction. The evaluation was a lab-based systematic comparison. We tested some individual and combined factors of the FFUIM. The performance indicator used was Mean Average Precision (MAP), and we used the ranking of images in the entire data set to compute the MAP for every experiment.

4.1 Experimental Setup

The ImageCLEFphoto2007 collection [3] was used, which consists of 20,000 real life images and 60 query topics. We applied colour feature HSV to all of the images. The City block distance (a special case of Minkowski distance family) was used to compute the distance between query images and object images.

Two Fusion Approaches. We used two fusion approaches to support two different RF scenarios. Firstly, the vector space model (VSM) [5] was deployed for positive relevance feedback only. By adding the weighting scheme of the FFUIM into the VSM, the approach is represented by:

$$D_{VSM} = \sum_i (d_{ij}/W_z), \tag{4}$$

where the D_{VSM} is the sum of the distance value between a query (containing i positive examples) and an object image j. W_z can be one of the three factors' weight W_o,W_p,W_f, or any combination weight of all three factors, depending upon which factor or combined factors is/are being tested.

Secondly, because the VSM in [5] only uses positive RF, we applied k-nearest neighbours (k-NN) for both positive and negative relevance feedback [5]. Here, by taking into account the weighting scheme, k-NN is given by:

$$D_{KNN} = \frac{\sum_{i \in N} (d_{ij}/W_z + \varepsilon)^{-1}}{\sum_{i \in P} (d_{ij}/W_z + \varepsilon)^{-1} + \varepsilon}, \tag{5}$$

[4] Please see more detail in footnote 1 and 2.

where D_{KNN} is the distance value between an object image j with all the example images (positive and negative) in the query. ε is a small positive number (e.g. 0.00001) to avoid division by zero. N and P denote the sets of positive and negative images in the query.

Two Interaction Approaches. Our experiments used two interaction approaches: pseudo RF and a method we call simulated user RF.

Firstly, pseudo RF was applied - a method widely used in information retrieval. Here there is no user interaction functionality with the RF approach. The system automatically takes the top three and bottom three images from the ranked last iteration search result of each query as positive and negative examples, respectively, to expand the current queries. The reason we take the bottom three images as negative feedback to expand the current queries is because, from our previous experiment, this approach outperforms the use of randomly chosen negative examples.

Secondly, so-called simulated user RF was used. This approach uses three truly relevant images from the top ranked results of each query and three non-relevant images from the bottom as tested against the official relevance judgments file. We derive this method to provide an automatic means of feedback which is closer to real user behavior. The reason we limit feedback to three positive images and three negative ones is because we want to make the experimental results more comparable with equal numbers of image examples in the queries.

For consistency of the two approaches, we used three image examples in each original query and each of the RF iterations. Further, we limited the number of iterations to be three, where iteration one is the search by original queries without RF, and iterations two and three are with RF. The time and relevance region factors are applied to all the queries on every iteration, whilst the relevance level and frequency factor is applied only to the latest query.

4.2 Experimental Results

Our experiment has tested the performance of 16 interaction settings of the FFUIM, which includes four profiles of OM (time factor): flat profile, increasing profile, current profile, decreasing profile, these profiles combined with the relevance level factor, and the above combinations joint with the frequency factor. Each of the 16 settings was tested using positive RF only as well as positive and negative RF (relevance region factor). The models have been tested against a large image collections and two interaction approaches as previously described. The following insights and analysis has been made, by doing statistical significance tests (the Wilcoxon signed ranks test with $\alpha = 0.05$):

Firstly, simulated user RF has better performance than pseudo RF. Secondly, with the pseudo RF approach, accuracy falls with increasing iterations. Thirdly, under simulated user RF approach, the performance clearly improves with each search iteration for all the results.

Apart from these generic insights, other results vary depending on the different settings and iterations. Since iteration three is the last iteration in our experiment and the weights should show more effect on the results, and, in addition, the simulated user RF outperforms pseudo RF and is closer to the real search scenario, we have undertaken further detailed analysis of the simulated RF at iteration three based on different search settings as follows:

Comparing the four profiles of the Ostensive Model (time factor). For the positive examples only setting, the decreasing and current profiles show consistently good performance, then the flat profile outperforms the increasing profile in most tests; for the both positive and negative example setting, the decreasing, flat and increasing profiles are not significantly different, but the current profile shows statistically worse performance than the other three profiles. The results do not show the same observation as previous OM studies, namely that the latest RF expresses best the user's information needs. This may be because the relevance judgement file was developed against the original query that is the oldest RF iteration. Thus the decreasing profile performs consistently well in different circumstances. These models need further testing in a real as opposed to simulated CBIR search environment.

With or without relevance level factor. In all of the tests, the relevance level when combined with the OM is not significantly different to the OM alone. This factor also needs further testing under a real user as opposed to simulated user evaluation.

With or without frequency factor. The frequency factor when combined with the other factors does not lead to significantly better performance than the factors without frequency factor. This may be because the limited number of search iterations means that the frequency weight has little impact. In addition, our definition of the frequency factor is that the latest query images are more important, which is different from the relevance judgement file that was created based on the original queries. This result may be clearer when we run further iterations of the experiment, or even under a real as opposed to simulated user evaluation.

Positive examples only and both positive and negative examples (relevance region factor). The use of both positive and negative example RF with k-NN approach performs significantly better than only positive example RF with VSM approach. The promising result encourages us to include the negative functionalities to our future visual search system, and then we need to think about how to deliver these functionalities to users through the interface.

5 Conclusion and Future Work

In an effort to alleviate the limitations of current user interaction models and to find a UI model to deliver a better interaction and search accuracy for CBIR, we have proposed a new four-factor user interaction model based on relevance region, relevance level, time and frequency. We have also empirically investigated different settings of the proposed model.

The following main observations have been made from the evaluation results: (1) bringing the user into the loop will enhance CBIR; (2) allowing both positive and negative feedback improves search performance; (3) combining the relevance level and frequency factor with other factors will make the user interaction model more usable and may well improve the search accuracy.

This work will be a foundation for developing more effective user interaction systems for CBIR. We have developed a visual content-based image search system, so that we can carry out real as opposite to simulated user experiments to evaluate the usefulness and effectiveness of the different settings of the FFUIM model. We are using a series of quantitative performance indicators, such as scores from questionnaires, precision of actual search results, time and number of clicks taken to complete the task, etc. Early results of the user study are under review and detailed analysis is underway.

References

1. Campbell, I.: Interactive evaluation of the ostensive model using a new test collection of images with multiple relevance assessments. Journal of Information Retrieval 2(1) (2000)
2. Dunlop, M.D.: The effect of accessing nonmatching documents on relevance feedback. ACM Transactions on Information Systems (TOIS) 15(2), 137–153 (1997)
3. Grubinger, M., Clough, P., Müller, H., Deselaers, T.: The iapr tc-12 benchmark: A new evaluation resource for visual information systems. In: Proceedings of International Workshop OntoImage 2006 Language Resources for Content-Based Image Retrieval, pp. 13–23 (2006)
4. Müller, H., Müller, W., Marchand-Maillet, S., Pun, T.: Strategies for positive and negative relevance feedback in image retrieval. In: Proceedings of the International Conference on Pattern Recognition (ICPR 2000), Barcelona, Spain, September 2000, vol. 1, pp. 1043–1046 (2000)
5. Pickering, M.J., Rüger, S.: Evaluation of key frame-based retrieval techniques for video. Computer Vision and Image Understanding 92(2-3), 217–235 (2003)
6. Ruthven, I., Lalmas, M., van Rijsbergen, K.: Incorporating user search behaviour into relevance feedback. Journal of the American Society for Information Science and Technology 54(6), 528–548 (2003)
7. Saracevic, T.: Relevance reconsidered. In: Proceedings of the Second Conference on Conceptions of Library and Information Science (CoLIS 2), Copenhagen,Denmark, October 1996, pp. 210–218 (1996)
8. Spink, A., Greisdorf, H., Bateman, J.: From highly relevant to not relevant: examining different regions of relevance. Information Processing Management 34(5), 599–621 (1998)
9. Urban, J., Jose, J.M., van Rijsbergen, K.: An adaptive technique for content-based image retrieval. Multimedia Tools and Applications 31, 1–28 (2006)
10. Zhou, X.S., Huang, T.S.: Relevance feedback in image retrieval: A comprehensive review. Multimedia Systems 8(6), 536–544 (2003)

Predicting Query Performance by Query-Drift Estimation

Anna Shtok[1], Oren Kurland[1], and David Carmel[2]

[1] Faculty of Industrial Engineering and Management, Technion, Haifa 32000, Israel
annabel@tx.technion.ac.il, kurland@ie.technion.ac.il
[2] IBM Haifa Research Labs, Haifa 31905, Israel
carmel@il.ibm.com

Abstract. Predicting *query performance*, that is, the effectiveness of a search performed in response to a query, is a highly important and challenging problem. Our novel approach to addressing this challenge is based on estimating the potential amount of *query drift* in the result list, i.e., the presence (and dominance) of aspects or topics not related to the query in top-retrieved documents. We argue that query-drift can potentially be estimated by measuring the *diversity* (e.g., standard deviation) of the retrieval scores of these documents. Empirical evaluation demonstrates the prediction effectiveness of our approach for several retrieval models. Specifically, the prediction success is better, over most tested TREC corpora, than that of state-of-the-art prediction methods.

Keywords: query-performance prediction, query drift, score distribution.

1 Introduction

Many information retrieval (IR) systems suffer from a radical variance in performance when responding to users' queries. Even for systems that succeed very well on average, the quality of results returned for some of the queries is poor [1]. Thus, it is desirable that IR systems will be able to identify "difficult" queries in order to handle them properly.

We present a novel approach to query-performance prediction that is based on estimating the potential amount of *query drift* in the *result list* — the documents most highly ranked in response to the query. That is, the presence and dominance of non-query-related aspects or topics manifested in documents in the list.

As it turns out, we potentially do not need to directly identify query-drift, rather we can use a proxy for its estimation. Specifically, using insights from work on pseudo-feedback-based query expansion [2] we argue that high standard deviation of retrieval scores in the result list correlates with reduced query-drift, and consequently, with improved effectiveness. Empirical evaluation demonstrates the prediction-effectiveness of our predictor for several retrieval methods, specifically, with respect to that of state-of-the-art predictors.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 305–312, 2009.

2 Related Work

Pre-retrieval query-performance prediction methods [3] analyze the query expression. However, the (short) query alone is often not expressive enough for reliable prediction [3]. The most effective prediction approaches employ post-retrieval analysis of the *result list* — the documents most highly ranked in response to the query. In what follows we discuss three such prominent paradigms.

The *clarity* prediction paradigm [4] is based on measuring the "focus" (clarity) of the result-list *with respect* to the corpus by computing different forms of their "distance" [5,6,7]. In Sect. 4 we show that our predictor is more effective than the clarity measure [4] over most tested collections.

Different notions of the *robustness* (e.g., with respect to document and query perturbations), and cohesion, of the result list [8,9,10,11,12] were shown to indicate query performance. Our proposed predictor, which measures the diversity of retrieval scores in the result list, can be thought of as a surrogate for estimating robustness with respect to document perturbations [9,10] — small *random* document perturbations are unlikely to result in major changes to documents' retrieval scores, and hence, are unlikely to significantly change the result list if retrieval scores are quite spread.

Work on analyzing *retrieval-scores distributions* to predict query performance showed that (i) the highest retrieval score [13], (ii) the difference between retrieval-scores produced in a query-independent and a query-dependent manner [14], and (iii) the extent to which similar documents receive similar retrieval scores [15] can indicate query performance. These techniques are complementary to ours. A state-of-the-art predictor, *Weighted Information Gain* (WIG) [12], measures the divergence between the mean retrieval score of top-ranked documents and that of the entire corpus. In contrast, our predictor essentially computes the divergence between the retrieval scores of top-ranked documents and that of a *pseudo non-relevant document* that exhibits a relatively high query-similarity. We demonstrate the merits of our predictor with respect to WIG in Sect. 4.

3 Prediction Framework

Let q, d, $\mathcal{D}$ and $\mathcal{M}$ be a query, document, corpus, and retrieval method, respectively. We use $Score(d)$ to denote the retrieval score assigned to d in response to q by $\mathcal{M}$. Our goal is to devise an estimate (predictor) for the *effectiveness* of the ranking induced by $\mathcal{M}$ over $\mathcal{D}$ in the *absence* of *relevance judgment* information. The estimated effectiveness is the *query performance* we attribute to $\mathcal{M}$ with respect to q. The methods we present utilize the result list $\mathcal{D}_q^{[k]}$ of the k documents that are the most highly ranked; k is a free parameter that is fixed to some value prior to retrieval (and prediction) time. As in many retrieval paradigms, we assume that $\mathcal{D}_q^{[k]}$ is composed of the documents that exhibit the highest (non-zero) surface-level similarity to q.

3.1 Estimating Query Drift

We refer to non-relevant documents in $\mathcal{D}_q^{[k]}$ as *misleaders* because they "mislead" the retrieval method into "believing" that they are relevant as they exhibit relatively high query-similarity. Misleaders are usually dominated by non query-related aspects (topics) that "drift away" from those represented by q [2].

As it turns out, we can potentially identify (at least) one (pseudo) misleader. Work on pseudo-feedback-based query expansion often uses a *centroid* representation, $Cent(\mathcal{D}_q^{[k]})$, of the list $\mathcal{D}_q^{[k]}$ as an expanded "query model" [16,17]. While using *only* the centroid yields poor retrieval performance [16,18,19], anchoring it to the query q via interpolation [18,19] yields improved performance, leading to the conclusion that the centroid manifests query drift [2]. Thus, $Cent(\mathcal{D}_q^{[k]})$ could be viewed as a prototypical misleader as it exhibits (some) similarity to the query by virtue of the way it is constructed (from documents in $\mathcal{D}_q^{[k]}$), but this similarity is dominated by non-query-related aspects that lead to query drift.

The degree of relevance of $Cent(\mathcal{D}_q^{[k]})$ to q is presumed by the retrieval method $\mathcal{M}$ to be correlated with its retrieval score, $\mu \stackrel{def}{=} Score(Cent(\mathcal{D}_q^{[k]}))$. In fact, we need not directly compute μ, because the mean retrieval score of documents in $\mathcal{D}_q^{[k]}$, $\hat{\mu} \stackrel{def}{=} \frac{1}{k}\sum_{d \in \mathcal{D}_q^{[k]}} Score(d)$, corresponds in several retrieval methods to the retrieval score, μ, of some centroid-based representation of $\mathcal{D}_q^{[k]}$. (We show that in Sect. 3.2). Thus, $\hat{\mu}$ represents the retrieval score of a prototypical misleader.

Estimates of Retrieval Effectiveness. Documents with retrieval scores (much) higher than $\hat{\mu}$, the score of a prototypical misleader, are potentially less probable to manifest query drift, and hence, be misleaders. Such documents could be considered as exhibiting positive ("+") *query-commitment* (QC). We therefore hypothesize that high divergence from $\hat{\mu}$ of the retrieval scores of these documents correlates with improved retrieval effectiveness. Since retrieval scores are query dependent, we normalize the divergence with respect to the retrieval score of a *general* prototypical non-relevant document, namely, the corpus. (We assume that the corpus can be represented as a single "pseudo" document, e.g., by using a centroid representation.) The resultant positive ("+") normalized-query-commitment (NQC) estimate is:

$$NQC_{+}(q, \mathcal{M}) \stackrel{def}{=} \frac{1}{Score(\mathcal{D})} \sqrt{\frac{1}{k} \sum_{d \in \mathcal{D}_q^{[k]} : Score(d) \geq \hat{\mu}} (Score(d) - \hat{\mu})^2} \; .$$

If we assume that there are only a few *relevant* documents in the corpus that yield "reasonable" query similarity, then a small overall number of documents exhibiting "reasonable" query-similarity can potentially indicate a small number of misleaders. The lower the retrieval score of a document is with respect to $\hat{\mu}$, the less we consider it to exhibit "reasonable" query-similarity (i.e., query-commitment). Hence, we hypothesize that the overall number of misleaders decreases (and hence, retrieval effectiveness increases) with increased (normalized) negative ("-") query-commitment measured by:

$$NQC_{-}(q,\mathcal{M}) \overset{def}{=} \frac{1}{Score(\mathcal{D})}\sqrt{\frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}:Score(d)<\hat{\mu}}(Score(d)-\hat{\mu})^2} \ .$$

We integrate the NQC_{+} and NQC_{-} measures to yield our main query-performance predictor, NQC, the (normalized) *standard deviation* of the retrieval scores in $\mathcal{D}_q^{[k]}$:

$$NQC(q,\mathcal{M}) \overset{def}{=} \sqrt{NQC_{+}(q,\mathcal{M})^2 + NQC_{-}(q,\mathcal{M})^2} = \frac{\sqrt{\frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}}(Score(d)-\hat{\mu})^2}}{Score(\mathcal{D})};$$

this measure has an appealing geometric interpretation exemplified in Fig. 1.

3.2 Use Case: Language Modeling Framework

The proposed performance-prediction measures can be employed with retrieval methods that estimate relevance based on surface-level document-query similarities. Here, we focus on the language modeling framework [21].

Let $p(w|d)$ be the probability assigned to term w by a (smoothed) unigram language model induced from document d. The commonly-used *query likelihood* (QL) retrieval method [20] scores document d in response to query $q = \{q_i\}$ by

$$Score_{QL}(d) = \sum_{q_i} \log p(q_i|d) \ . \qquad (1)$$

To compute the corpus retrieval score $Score_{QL}(\mathcal{D})$, we treat $\mathcal{D}$ as the document that results from concatenating all documents in $\mathcal{D}$; the order of concatenation has no effect, since we use unigram language models.

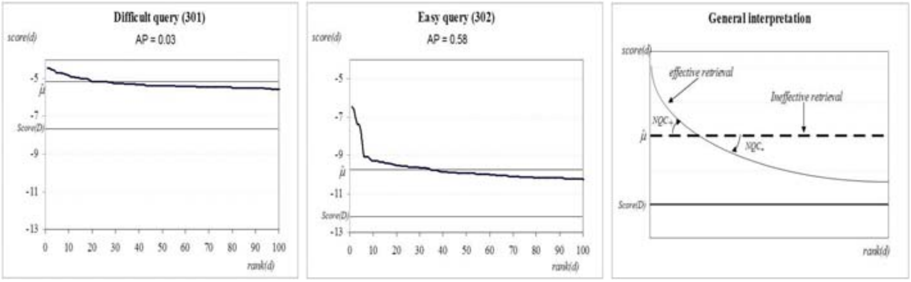


Fig. 1. Geometric interpretation of NQC. The two leftmost graphs present retrieval-scores curves for "difficult" and "easy" queries chosen by average-precision (AP) performance (query-likelihood model [20], ROBUST benchmark). Right: the shift between these two scenarios amounts to clockwise rotation of the retrieval-scores line.

The Centroid We stated in Sect. 3.1 that the mean retrieval score ($\hat{\mu}$) of documents in $\mathcal{D}_q^{[k]}$ corresponds to the retrieval score of a centroid-based representation of $\mathcal{D}_q^{[k]}$. We now demonstrate this correspondence for the query likelihood model.

Proposition 1. *The mean of the QL-retrieval-scores of documents in $\mathcal{D}_q^{[k]}$ is the QL score of a geometric-centroid language-model-based representation of $\mathcal{D}_q^{[k]}$.*

Proof. Let $\hat{\mu} = \frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}} Score_{QL}(d)$. By definition, $\hat{\mu} = \frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}}\sum_{q_i} \log p(q_i|d)$. We can re-arrange the summation and write $\hat{\mu} = \sum_{q_i}\frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}} \log p(q_i|d)$ $= \sum_{q_i} \log \sqrt[k]{\prod_{d\in\mathcal{D}_q^{[k]}} p(q_i|d)}$. We define $p(w|Cent(\mathcal{D}_q^{[k]})) \overset{def}{=} \sqrt[k]{\prod_{d\in\mathcal{D}_q^{[k]}} p(w|d)}$ — a language model (modulo normalization details) that corresponds to the geometric-centroid of language models of documents in $\mathcal{D}_q^{[k]}$; similar centroid was used in recent work on cluster-based retrieval [22]. By Eq. 1, $Score_{QL}(Cent(\mathcal{D}_q^{[k]})) = \hat{\mu}$.

The connection between the mean retrieval score of documents in $\mathcal{D}_q^{[k]}$ and the retrieval score of a centroid of $\mathcal{D}_q^{[k]}$ holds for other retrieval functions that are linear in features. For example, let $\boldsymbol{x}$ be the vector-space representation of text x. Now, if $Cent(\mathcal{D}_q^{[k]}) \overset{def}{=} \frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}} \boldsymbol{d}$ is the algebraic-centroid of $\mathcal{D}_q^{[k]}$, and the inner product is used as a retrieval function, then $\hat{\mu} \overset{def}{=} \frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}} <\boldsymbol{q},\boldsymbol{d}> = <\boldsymbol{q}, \frac{1}{k}\sum_{d\in\mathcal{D}_q^{[k]}} \boldsymbol{d}> = <\boldsymbol{q}, Cent(\mathcal{D}_q^{[k]})>$.

4 Evaluation

We evaluate prediction quality by measuring Pearson's [7] and Kendall's$-\tau$ [1] correlation between the actual performance (average precision at cutoff 1000), and accordingly, induced ordering, of queries in a given set (as determined by using relevance judgments), and the values (and accordingly, induced ordering) assigned to these queries by a performance predictor. For both measures, higher correlation values indicate increased prediction quality. All correlation numbers that we report are statistically significant at a 95% confidence level.

4.1 Experimental Setup

We conducted experiments on TREC collections used in previous query-performance-prediction studies [8,10,23,15]: (i) WT10G (topics 451-550), (ii) ROBUST (disks 4&5-CR, topics 301-450, 601-700), (iii) TREC123 (disks 1&2, topics 51-200), (iv) TREC4 (disks 2&3, topics 201-250), and (v) TREC5 (disks 2&4, topics 251-300).

We use the titles of TREC topics for queries, except for the TREC4 case, where no titles are provided, and hence, topic descriptions are used. We applied tokenization, Porter-stemming, and stopword removal (using the INQUERY list) to all data via the Lemur toolkit (www.lemurproject.org), which was also used for retrieval. The query likelihood model [20] described in Sect. 3.2 served as the

Table 1. Comparison of NQC with state-of-the-art predictors. The best result per collection and evaluation measure is boldfaced.

		Pearson			Kendall's$-\tau$		
Corpus	#topics	Clarity	WIG	NQC	Clarity	WIG	NQC
WT10G	100	0.331	0.376	**0.527**	0.285	0.3	**0.303**
ROBUST	249	0.513	0.543	**0.563**	0.411	0.386	**0.419**
TREC123	150	0.462	**0.624**	0.376	0.351	**0.437**	0.273
TREC4	50	0.478	0.543	**0.556**	0.389	**0.489**	0.414
TREC5	50	**0.441**	0.297	0.431	**0.312**	0.253	0.3

retrieval model. (We used Dirichlet smoothing with the smoothing parameter set to 1000 following previous recommendations [24].)

We compare the prediction quality of NQC with that of two state-of-the-art predictors: Clarity [4] and WIG [12]. Clarity measures the KL divergence between a relevance language model (RM1) [17] constructed from the result-list $\mathcal{D}_q^{[k]}$ and the corpus model. We use Lemur's Clarity implementation.[1]

WIG was originally proposed in the MRF framework [25]. If term-dependencies are not used, MRF reduces to the query likelihood model with unigram language models. (It was noted that WIG is very effective with such implementation [23].) In this case, $WIG(q, QL) \overset{def}{=} \frac{1}{k} \sum_{d_i \in \mathcal{D}_q^{[k]}} \frac{1}{\sqrt{|q|}} (Score_{QL}(d_i) - Score_{QL}(\mathcal{D}))$.

Following experiments (results omitted due to space considerations) with different values of k, the number of documents in the result-list $\mathcal{D}_q^{[k]}$, we set its value to 100 for both our NQC measure and the Clarity predictor, and to 5 for WIG (which is in accordance with previous recommendations [23]).[2]

4.2 Experimental Results

The results in Table 1 show that NQC predicts query-performance very well over most collections. Specifically, NQC outperforms each of the baselines, WIG and Clarity, over three out of the five collections with respect to both evaluation measures. We attribute the relatively low prediction quality of NQC for TREC123 to the fact that TREC123 has extremely high average number of relevant documents per topic with respect to the other collections. Indeed, if NQC is employed for TREC123 over a much larger result-list, then prediction success can improve up to a Pearson correlation of 0.7; the same holds for WIG.

Table 2 shows that both NQC_+ and NQC_- that are integrated by NQC are effective performance predictors. (Note the relatively high correlation numbers.) We also see that NQC is more effective than NQC_+ and NQC_- over three collections with respect to both evaluation measures. These findings support the importance of considering both NQC_+ and NQC_- as described in Sect. 3.

[1] We found that *clipping* RM1 so as to use 100 terms yields much better prediction-quality than using all terms as previously suggested [6].

[2] The prediction quality of (i) the Clarity measure is highly stable with respect to k, (ii) the WIG measure is in general optimal for low values of k (specifically, $k = 5$), and (iii) our NQC measure is in general quite stable for $k \in [80 - 500]$.

Table 2. Prediction quality of NQC sub-components: NQC$_+$ and NQC$_-$. Best result per collection and evaluation measure is boldfaced.

		Pearson			Kendall's$-\tau$		
Corpus	#topics	NQC$_+$	NQC$_-$	NQC	NQC$_+$	NQC$_-$	NQC
WT10G	100	**0.531**	0.479	0.527	**0.326**	0.274	0.303
ROBUST	249	0.560	0.519	**0.563**	0.416	0.397	**0.419**
TREC123	150	0.307	**0.48**	0.376	0.236	**0.336**	0.273
TREC4	50	0.526	**0.614**	0.556	0.388	**0.471**	0.414
TREC5	50	**0.491**	0.287	0.431	**0.333**	0.297	0.300

Table 3. Prediction quality (Pearson correlation) of NQC for the vector space model (with the cosine measure), Okapi, and the language model (LM) approach used so far

	Vector space	Okapi	LM
WT10G	0.407	0.311	0.527
ROBUST	0.535	0.603	0.563
TREC123	0.609	0.369	0.376
TREC4	0.664	0.578	0.556
TREC5	0.448	0.423	0.431

Table 3 presents the Pearson correlation for using NQC with the cosine measure in the vector space and with the Okapi BM25 method[3]. The (relatively high) correlation for both methods, which sometimes transcends that for the language-model approach used insofar, attests to the general effectiveness of NQC as a query-performance predictor.

5 Summary

We presented a novel approach to predicting query performance that is based on estimating the potential amount of *query drift* in the list of top-retrieved documents using the standard deviation of their retrieval scores. Empirical evaluation demonstrates the effectiveness of our predictor with several retrieval methods.

Acknowledgments. This paper is based upon work supported in part by Google's and IBM's faculty awards. Any opinions, findings, and conclusions or recommendations expressed are those of the authors and do not necessarily reflect those of the sponsors.

References

1. Voorhees, E.M.: Overview of the TREC 2004 Robust Retrieval Track. In: Proceedings of TREC-13 (2004)
2. Mitra, M., Singhal, A., Buckley, C.: Improving automatic query expansion. In: Proceedings of SIGIR, pp. 206–214 (1998)
3. Hauff, C., Hiemstra, D., de Jong, F.: A survey of pre-retrieval query performance predictors. In: Proceedings of CIKM, pp. 1419–1420 (2008)

[3] Since cosine scores are embedded in the unit sphere, normalization with the corpus retrieval-score is redundant (and, degrades prediction quality); it is therefore not employed. To avoid underflow issues caused by the document-length-normalization in Okapi, we use the centroid of all documents in the corpus to represent it. Lemur's implementation of both methods is used with default parameter settings.

4. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: Proceedings of SIGIR, pp. 299–306 (2002)
5. Amati, G., Carpineto, C., Romano, G.: Query difficulty, robustness and selective application of query expansion. In: McDonald, S., Tait, J.I. (eds.) ECIR 2004. LNCS, vol. 2997, pp. 127–137. Springer, Heidelberg (2004)
6. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Precision prediction based on ranked list coherence. Information Retrieval 9(6), 723–755 (2006)
7. Carmel, D., Yom-Tov, E., Darlow, A., Pelleg, D.: What makes a query difficult? In: Proceedings of SIGIR, pp. 390–397 (2006)
8. Yom-Tov, E., Fine, S., Carmel, D., Darlow, A.: Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In: Proceedings of SIGIR, pp. 512–519 (2005)
9. Vinay, V., Cox, I.J., Milic-Frayling, N., Wood, K.R.: On ranking the effectiveness of searches. In: Proceedings of SIGIR, pp. 398–404 (2006)
10. Zhou, Y., Croft, W.B.: Ranking robustness: a novel framework to predict query performance. In: Proceedings of CIKM, pp. 567–574 (2006)
11. Aslam, J.A., Pavlu, V.: Query hardness estimation using Jensen-Shannon divergence among multiple scoring functions. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECIR 2007. LNCS, vol. 4425, pp. 198–209. Springer, Heidelberg (2007)
12. Zhou, Y., Croft, W.B.: Query performance prediction in web search environments. In: Proceedings of SIGIR, pp. 543–550 (2007)
13. Tomlinson, S.: Robust, Web and Terabyte Retrieval with Hummingbird Search Server at TREC 2004. In: Proceedings of TREC-13 (2004)
14. Bernstein, Y., Billerbeck, B., Garcia, S., Lester, N., Scholer, F., Zobel, J.: RMIT university at TREC 2005: Terabyte and robust track. In: Proceedings of TREC-14 (2005)
15. Diaz, F.: Performance prediction using spatial autocorrelation. In: Proceedings of SIGIR, pp. 583–590 (2007)
16. Rocchio, J.J.: Relevance feedback in information retrieval. In: Salton, G. (ed.) The SMART Retrieval System: Experiments in Automatic Document Processing, pp. 313–323. Prentice Hall, Englewood Cliffs (1971)
17. Lavrenko, V., Croft, W.B.: Relevance-based language models. In: Proceedings of SIGIR, pp. 120–127 (2001)
18. Zhai, C., Lafferty, J.D.: Model-based feedback in the language modeling approach to information retrieval. In: Proceedings of CIKM, pp. 403–410 (2001)
19. Abdul-Jaleel, N., Allan, J., Croft, W.B., Diaz, F., Larkey, L., Li, X., Smucker, M.D., Wade, C.: UMASS at TREC 2004 — novelty and hard. In: Proceedings of TREC-13 (2004)
20. Song, F., Croft, W.B.: A general language model for information retrieval (poster abstract). In: Proceedings of SIGIR, pp. 279–280 (1999)
21. Croft, W.B., Lafferty, J. (eds.): Language Modeling for Information Retrieval. Information Retrieval Book Series, vol. 13. Kluwer, Dordrecht (2003)
22. Liu, X., Croft, W.B.: Evaluating text representations for retrieval of the best group of documents. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 454–462. Springer, Heidelberg (2008)
23. Zhou, Y.: Retrieval Performance Prediction and Document Quality. PhD thesis, University of Massachusetts (September 2007)
24. Zhai, C., Lafferty, J.D.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: Proceedings of SIGIR, pp. 334–342 (2001)
25. Metzler, D., Croft, W.B.: A Markov random field model for term dependencies. In: Proceedings of SIGIR, pp. 472–479 (2005)

What's in a Link?
From Document Importance to Topical Relevance

Marijn Koolen[1] and Jaap Kamps[1,2]

[1] Archives and Information Studies, University of Amsterdam, The Netherlands
[2] ISLA, University of Amsterdam, The Netherlands

Abstract. Web information retrieval is best known for its use of the Web's link structure as a source of evidence. Global link evidence is by nature query-independent, and is therefore no direct indicator of the topical relevance of a document for a given search request. As a result, link information is usually considered to be useful to identify the 'importance' of documents. Local link evidence, in contrast, is query-dependent and could in principle be related to the topical relevance. We analyse the link evidence in Wikipedia using a large set of ad hoc retrieval topics and relevance judgements to investigate the relation between link evidence and topical relevance.

1 Introduction

Web information retrieval is best known for its use of the Web's link structure as a source of evidence. PageRank [11] is a query-independent algorithm that measures document importance on a global level and is not concerned with a topical relation to the query at hand. The alternative is to analyse the link structure of local sets of documents—e.g., the initial text-based results—to identify topically authoritative pages for broad topics [1, 7]. What is the value of links in topic relevance tasks? This question was addressed by constructing an IR test collection during the 1999 Small Web Task at TREC [13], where participants tried to answer the question "whether hyperlink information could be used to improve ad hoc retrieval effectiveness" [4]. The results from the experiments failed to demonstrate the value of link information for ad hoc retrieval.

Arguably, the notion of what is relevant for typical Web searches is different from the traditional IR interpretation of a document containing text relevant to a precisely defined information need. New Web-oriented tasks were designed to better reflect Web search behaviour. In these Web tasks, the goal was to identify entry pages to particular sites (in the case of home page finding and topic distillation) or another important document (in the case of named page finding). These tasks also dictated a different notion of relevance [12]. The experiments showed that, although links were not effective for singling out the documents with topically relevant textual content, they are useful for locating the documents that are important for these Web-oriented tasks. This leads to our main research question:

- To what extent is link evidence related to the importance of documents, and to the topical relevance of documents?

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 313–321, 2009.

Here, global and local link evidence seem to play different roles. Links are also directed, and link evidence is typically used for the documents they point *to*, i.e., inlinks. Thinking of incoming links as some sort of vote, inlinks are attractive to measure document importance. However, insofar as a link is evidence that the two documents it connects are topically related, the direction of the link seems not to matter. Topical relatedness works both ways.

In this light, Wikipedia is an interesting data source to investigate the value of links. It is one of the most popular web sites and, being an encyclopedia, it contains entries on single topics, that are densely linked to related content. It is also a natural source for informational search, where it makes sense to study topical relevance aspects of links. Moreover, an extensive IR test collection based on Wikipedia is available thanks to the INEX Ad hoc Tracks of 2006 to 2007. Clearly, the Wikipedia differs considerably from the Web at large, and even the links in Wikipedia are different. We make no particular claims on the representativeness of the Wikipedia for the general Web. Still, the same link-related phenomena (global and local, incoming and outgoing links) are present, and looking at the Wikipedia allows us to study them in great detail. The INEX Ad hoc test collection allows us to study the impact of query-dependent and query-independent link evidence with respect to the topical relevance of retrieval results. In fact, because the INEX test collections are constructed to study the effectiveness of focused retrieval, we have exact information on where and how much relevant text is in each article.

We will first analyse the effectiveness of link evidence for ranking retrieval results, addressing the questions:

- What are the characteristics of Wikipedia link structure?
- How do global and local link evidence impact retrieval effectiveness?
- How do incoming, outgoing and undirected links impact retrieval effectiveness?

Then, we look at how related the different types of link evidence are:

- How do incoming, outgoing and undirected link evidence correlate?
- How is link evidence related to the amount of relevant text in articles?

The rest of this paper is structured as follows. In Section 2 we discuss related work. After discussing the experimental data in Section 3, we compare the different types of link evidence in a retrieval setting in Section 4. Then, in Section 5 we analyse the relation between the different degrees structures and the amount of relevant text in articles. We draw conclusions in Section 6.

2 Related Work

In the TREC Web Tracks of 1999 to 2004, participants were unable to show the effectiveness of link evidence for general ad hoc retrieval [3]. However, it was argued that traditional ad hoc retrieval is very different from how people search on the Web. To study the value of link information, tasks closer to real Web search are required [4]. With tasks adjusted to Web search scenarios, link information proved highly beneficial [8, 10]. This difference in effectiveness of link evidence for these tasks indicates that link evidence does not reflect topical relevance.

Table 1. Link statistics of the Wikipedia collections. Local statistics are macro averages over 221 topics.

	Global					Local				
Degree	**min**	**max**	**mean**	**median**	**stdev**	**min**	**max**	**mean**	**median**	**stdev**
Indegree	0	74,937	20.63	4	282.94	0	48.83	3.17	1.14	6.65
Outdegree	0	5,098	20.63	12	36.70	0.04	21.01	3.17	2.34	3.37
Union	0	75,072	37.65	16	287.87	0.04	51.11	5.14	3.14	7.19
Intersection	0	1,488	3.62	2	9.10	0	14.68	1.20	0.44	2.15

Najork et al. [9] compared HITS authorities and hubs with several link-based ranking algorithms – PageRank, Indegree and Outdegree – and found that the choice of algorithm makes little difference on the effectiveness of link evidence. What does have a big impact is the *direction* in which the evidence is used. Although adding evidence based on outgoing links to a content-based retrieval baseline does lead to improvements, it is much less effective than evidence based on incoming links.

Kamps and Koolen [5] showed that in Wikipedia, indegree is related to topical relevance and found that incoming link evidence can be effective for ad hoc retrieval. Later, they found that, unlike in the Web, incoming and outgoing link evidence is equally effective for document retrieval on Wikipedia [6].

3 Wikipedia Link Structure

For the analysis, we use the INEX Wikipedia collection [2], containing 659,304 documents, and a set of 221 topics with relevance judgements from the INEX 2006–2007 Ad hoc Tracks. The union of the in- and outdegree is the undirected degree, or the total number of pages that a page is connected to. The intersection of in- and outdegree is the set of bidirectional links, where pages A and B link to each other. The graph contains 12.4M undirected links and 1.2M bidirectional links (9.5%). We also look at local link evidence—considering only links between the top 100 ranked pages for a given query.

Degree statistics are shown in Table 1. Looking at the global link structure, the maximum indegree is much higher than the maximum outdegree. The maximum and spread of the undirected degree are very similar to those of the indegree, but the median is more similar to that of the outdegree. The bidirectional degree is much lower because only a small proportion (9.5%) of the links are bidirectional. When we look at the local degrees, we see a similar pattern. The indegrees have a bigger spread than the outdegrees, with the undirected degrees having a maximum and spread close to those of the indegrees and a median closer but above that of the outdegrees.

The number of local links is of course smaller than in the whole link graph, but the link density is higher. Globally, a document is connected to 0.0057% of the collection on average, whereas in the local set, it is connected to 5.14%. We also look at the proportion of bidirectional links by looking at the fraction of intersection within union. This proportion is much higher in the local set (23.4%) than in the global set (9.5%). This can be explained, at least in part, by the higher link density in the local set. The nature of Wikipedia links may also play a role: the Wikipedia guidelines on linking [14] state that a link to another document should only be made when it is relevant to the

Table 2. Impact of link evidence on the INEX 2006 and 2007 Adhoc Track topics. Significance levels are 0.05 ($^{\circ}$), 0.01 ($^{\odot}$) and 0.001 ($^{\bullet}$), bootstrap, one-tailed.

		Global				**Local**			
	Run	**MAP**	**P@5**	**P@10**	**P@30**	**MAP**	**P@5**	**P@10**	**P@30**
Link only	*Indegree*	14.41	29.68	28.14	24.77	21.20	47.06	41.49	32.17
	Outdegree	13.56	25.70	25.29	24.34	21.46	44.07	41.09	32.96
	Union	14.05	27.96	27.33	24.18	22.26	47.15	42.31	33.92
	Intersection	14.36	30.95	27.56	24.66	20.45	44.43	39.28	30.71
Content only	*baseline*	30.65	55.57	48.91	35.87	30.65	55.57	48.91	35.87
Content+Link	*Indegree*	26.66	50.50	41.90	31.79	31.71$^{\bullet}$	59.00$^{\bullet}$	50.27	36.80°
	Outdegree	27.73	52.13	43.98	32.38	31.83$^{\bullet}$	56.47	49.82	37.12$^{\odot}$
	Union	27.51	50.86	43.89	32.08	32.09$^{\bullet}$	57.83°	50.50$^{\odot}$	37.53$^{\bullet}$
	Intersection	28.41	53.12	45.61	32.87	31.75$^{\bullet}$	57.83°	50.18	37.10$^{\odot}$

context. Thus, in a set of documents related to the same query, many documents will be related to each other and therefore cross-linked.

4 Link Evidence

In this section, we investigate the impact of link evidence on the effectiveness of ad hoc retrieval. After that, we use standard IR effectiveness measures to evaluate a baseline run using a language modelling framework and runs derived from the baseline but re-ranked 1) using only link evidence and 2) using a combination of content and link evidence.

4.1 Using Only Link Evidence

We show the results in Table 2 and will first discuss the impact of using only link evidence (and not content) for ranking. When re-ranking on global link evidence only, indegree leads to higher early precision than outdegree, which is consistent with the hypothesis that global link structure signals 'important' pages. With the union of the degrees, precision is lower than with indegree alone, but higher than with outdegrees. The intersection of the degrees leads to a higher early precision than the indegree. The intersection creates a symmetric link graph that is still query-independent and seems more effective than indegree alone. Compared to the content-only run, though, the global link degrees are nowhere near as effective.

Although still well below the content-only run, local link degrees give much higher scores than global degrees, indicating that by biasing the link evidence by considering only links between documents related to the query, link information becomes more 'semantic'. The indegrees lead to higher early precision than the outdegrees but, overall, the outdegrees lead to a better ranking. The undirected or union degrees give even higher scores, showing that both individual degrees contribute complementary evidence on the relevance of documents. The scores of the intersection of the degrees are somewhat lower than those of the other degrees, which is probably due to the fact that the number of bidirectional links in the local set is relatively low.

4.2 Combining Link and Content Evidence

We now look at re-ranking using the combination of content and link evidence, which we do by multiplying the retrieval score by a link degree score:

$$P_{\text{Degree}}(d) \propto 1 + \mathsf{Degree}(d) \tag{1}$$

The bottom left part of Table 2 shows the combination of the baseline with the global link evidence. It is clear that the global link evidence universally hurts the baseline performance. The impact of the outdegree is smaller than that of the indegree, which makes sense given the bigger spread of the indegrees (see Table 1). The impact of the union of the degrees is closer to that of the outdegree, whereas the small number of bidirectional links in the local set keep the negative impact of the global link evidence small. Both in isolation and in combination with content evidence, global link degrees fall short of the performance of the content-only baseline.

Using local evidence (bottom-right part of Table 2), both indegree and outdegree can significantly improve the baseline run. Although the indegree gives bigger improvements in early precision and the outdegree gives bigger improvements further down the ranking, at P@30, their overall improvements are very similar. Links in Wikipedia can be used effectively in both directions as evidence to re-rank retrieval results. We then expect that ignoring the direction of links and counting the number of connections to other documents in the local set will lead to even better performance. The scores indeed show further, albeit small, improvements. Precision at rank 5 is higher than for the outdegrees, and later and overall precisions are higher than with both in- and outdegrees. When we use only the smaller set of bidirectional links, the results are still surprisingly good. With less than a quarter of the total links, the intersection of the degrees gives the same performance boost as the in- and outdegrees individually.

To summarise, global link evidence may be an indicator of document importance, but fails to help locate topically relevant documents to a specific information need. Local link evidence fares much better. In isolation, it gives much better performance than global link evidence, although it cannot compete with content-based evidence. In combination with this content-only baseline, it does lead to improvements. In fact, this combination is effective, whether we use only incoming links, outgoing links or their union or intersection. This result supports our intuition that for topical relatedness, the direction of links is of no importance. But in- and outdegrees affect the ranking differently. In what way do incoming and outgoing link evidence differ from each other? We address this question in the next section.

5 Relation between Degrees

In this section we analyse the extent to which degrees are correlated to each other. The main difference found so far is between global and local link evidence. The differences between incoming, outgoing, undirected and bidirectional link degrees are relatively small. A simple explanation would be that all these degrees are strongly correlated. Incoming and outgoing link evidence are necessarily related in some way: a link between two documents is incoming link evidence for one document and not the other, and vice versa for outgoing link evidence.

Table 3. Rank correlations between global, top 100 and top 10 local degrees

	Global				**Top 100**				**Top 10**			
Degree	*In*	*Out*	*Union*	*Inter*	*In*	*Out*	*Union*	*Inter*	*In*	*Out*	*Union*	*Inter*
In	–	0.41	0.59	0.66	–	0.49	0.71	0.77	–	0.30	0.77	0.43
Out		–	0.83	0.46		–	0.82	0.58	0.13	–	0.47	0.24
Union			–	0.50			–	0.59	0.63	0.32	–	0.37
Inter				–				–	0.48	0.49	0.45	–

5.1 Correlation of Degrees

We computed rank correlations (Kendall's Tau) between the four degree types over the entire collection, and within the local set of retrieved results for all 221 topics (Table 3). Over the entire collection, the in- and outdegree are moderately correlated, and the undirected degree is very strongly correlated with outdegree and less strongly with the indegree. This means that, on a global level, the outdegree is the dominant factor in the undirected degree. The same holds for the bidirectional degree. The intersection correlates most strongly with the indegree. Over the local top 100 link graphs, using the average of the correlations of the 221 topics, the correlation between in- and outdegree is stronger, and thereby, their correlation with the union is more similar. An explanation might be the higher percentage of bidirectional links in the local sets. The overall correlations give a broad idea of the relationship between degrees. Given that most documents have a low in- and outdegree, the correlation is dominated by these low degrees while we are mostly interested in the other end with the highest degrees.

In Table 3 we also show the correlations between degrees over the top 10 results. That is, we take the top 10 results ranked by the column (say, indegree) and compare their ordering with how they are ranked by the row (say, outdegree). Note that over the top 10, the correlation is not symmetrical: the top 10 documents by indegree can be different from the top 10 documents by outdegree. Over the top 10, the rank correlation between indegree and outdegree is lower than over the top 100. The top 10 ranking by indegree corresponds better to their ranking by outdegree than the top 10 ranking by outdegree corresponds to their ranking by indegree. The average overlap between the two sets of top 10 documents is 4.7, thus each has 5.3 documents in the top 10 that are not in the top 10 of the other. Over the top 100, the outdegree correlates stronger with the undirected degree than the indegree, but over the top 10, it is the other way around. This is reflected in the precision scores in Table 2. The undirected degree has an early precision very similar to that of the indegree, while further down the results list, its precision is closer to that of the outdegree. The correlations with the intersection are much lower over the top than over the top 100, probably because of the lower degrees.

5.2 Correlation of Degree and Relevant Text Size

In Section 3, we saw that all four types of local link evidence show some relation to topical relevance, as evidenced by their positive effect on performance when combined with the content score. But not all documents are equally relevant. Some documents

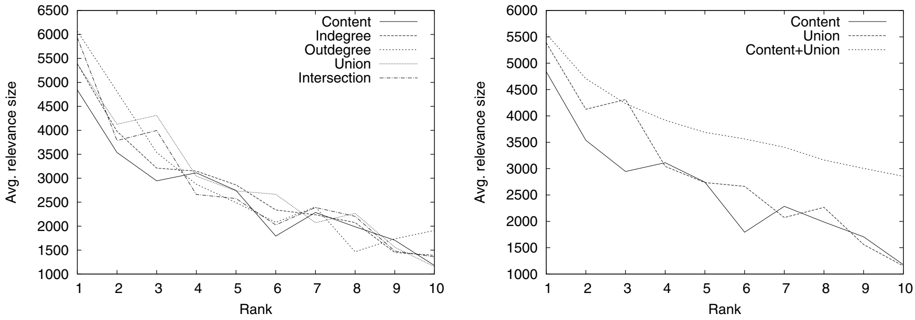


Fig. 1. The average amount of relevant text at ranks 1 to 10 for the retrieved relevant documents ranked by content or link degree

might be mostly off-topic and only mention the topic in a few sentences, while others might be fully on-topic and cover the topic exhaustively. For the INEX Ad hoc Track, assessors are asked to highlight in yellow all and only relevant text within each pooled document. This allows us to study the relation between link evidence and the amount of relevant text. We assume that documents that have more relevant text discuss the topic more exhaustively and are therefore more important to the topic.

Figure 1 shows the average amount of relevant text over the first 10 retrieved relevant documents when ranked by degree. The left-hand-side shows the content-only and link-only evidence. The right-hand-side shows the content and undirected link evidence and their combination. We see that the amount of relevant text decreases over rank for all types of evidence. The content-only evidence has the lowest amount of relevant text at rank 1, and the outdegree the highest. In the set of retrieved relevant documents, link evidence seems to be a good indicator of the amount of relevant text in a document. This makes sense if local link evidence is related to topical relevance. More links means more evidence of topical relevance, and thus more relevant to the topic. In the right-hand figure, it is clear that the undirected degree ranking has more relevant text at most ranks, especially at the first 3 ranks. Thus, although the content-only score is a better indicator of the relevance of a document—it has much higher precision and MAP scores in Table 2—the undirected link evidence seems a better indicator of the amount of relevant text in documents. The combination of both types of evidence has a large impact on the amount of relevant text found at all first 10 ranks. This indicates that the relevant articles are ranked more favourably, an important aspect that remains unnoticed by the standard evaluation measures.

6 Discussion and Conclusions

In this paper we investigated the relation between link evidence and topical relevance in Wikipedia. Our main aim was to find out to what extent link evidence is related to document importance and to topical relevance.

The local link structure is more dense than the global link structure and has a larger proportion of bidirectional links, making link evidence more symmetrical. Evidence based on incoming links gives better early precision, while outgoing link evidence gives better precision further down the ranking. Taking the union of these two degrees leads to further improvements, showing that in- and outdegrees contribute different information. At the local level, the different degrees all derived from the same link graph exhibit reasonably high correlations, indicating that they promote many of the same documents. However, this correlation is lower in the top of the in- and outdegree rankings. Given the substantial difference between documents in the top 10, in- and outdegree seem to promote different documents. The degrees can also help the internal ranking of the relevant documents by inducing a more favourable ranking in terms of the amount of relevant text in articles. One could think of notions of relevance for Web retrieval extending the traditional topical relevance, for example, by requiring pages to be both 'relevant' in the traditional sense, as well as 'important' or 'authoritative'. Such a view would impose an additional criterion on the topically relevant pages, which is supported by our analysis of the amount of relevant text in documents.

This paper is only a first step in understanding the value of link information. Wikipedia is different from the Web at large, including its links: the Web is much more heterogeneous and noisy, and the creation of Web links is not steered by clear guidelines, nor done for a single purpose. Nevertheless, the distinction between global and local evidence holds for the Web as well. But our analysis of links in Wikipedia has shown that the link structure contains valuable cues about topical relevance.

Acknowledgments. This research was supported by the Netherlands Organization for Scientific Research (NWO, grants # 612.066.513, 639.072.601, and 640.001.501).

References

[1] Carrière, S.J., Kazman, R.: Webquery: Searching and visualizing the web through connectivity. Computer Networks 29(8-13), 1257–1267 (1997)

[2] Denoyer, L., Gallinari, P.: The Wikipedia XML Corpus. SIGIR Forum 40(1), 64–69 (2006)

[3] Hawking, D.: Overview of the TREC-9 web track. In: The Ninth Text REtrieval Conference (TREC-9), pp. 87–102. NIST Special Publication 500-249 (2001)

[4] Hawking, D., Craswell, N.: Very large scale retrieval and web search. In: TREC: Experiment and Evaluation in Information Retrieval, ch. 9, MIT Press, Cambridge (2005)

[5] Kamps, J., Koolen, M.: The importance of link evidence in Wikipedia. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 270–282. Springer, Heidelberg (2008)

[6] Kamps, J., Koolen, M.: Is Wikipedia link structure different? In: Proceedings of the Second ACM International Conference on Web Search and Data Mining (WSDM 2009). ACM Press, New York (2009)

[7] Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. Journal of the ACM 46, 604–632 (1999)

[8] Kraaij, W., Westerveld, T., Hiemstra, D.: The importance of prior probabilities for entry page search. In: Proceedings of the 25th Annual International ACM SIGIR Conference, pp. 27–34. ACM Press, New York (2002)

[9] Najork, M., Zaragoza, H., Taylor, M.: Hits on the web: How does it compare? In: SIGIR 2007 (2007)

[10] Ogilvie, P., Callan, J.: Combining document representations for known-item search. In: Proceedings of the 26th Annual International ACM SIGIR Conference, pp. 143–150. ACM Press, New York (2003)

[11] Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project (1998)

[12] Saracevic, T.: Relevance: A review of and a framework for the thinking on the notion in information science. Journal of the American Society for Information Science 26, 321–343 (1975)

[13] TREC. Text-REtrieval Conference (2009), http://trec.nist.gov/

[14] Wikipedia. Linking (2009),
http://en.wikipedia.org/wiki/Wikipedia:Linking

Avoiding Bias in Text Clustering Using Constrained K-means and May-Not-Links

M. Eduardo Ares, Javier Parapar, and Álvaro Barreiro

IRLab, Department of Computer Science, University of A Coruña, Spain
{maresb,javierparapar,barreiro}@udc.es

Abstract. In this paper we present a new clustering algorithm which extends the traditional batch k-means enabling the introduction of domain knowledge in the form of Must, Cannot, May and May-Not rules between the data points. Besides, we have applied the presented method to the task of avoiding bias in clustering. Evaluation carried out in standard collections showed considerable improvements in effectiveness against previous constrained and non-constrained algorithms for the given task.

1 Introduction

Clustering [1] and classification [2] methods have been demonstrated as useful tools in several fields within computer science, like Data-Mining (DM) or Information Retrieval (IR). The need for methods for automatic data analysis has arisen when working with large collections of heterogeneous data, where doing it manually by experts was unfeasible. Even though the main difference between clustering and classification has been that the later is performed without any prior knowledge of the data, adding some domain knowledge to the clustering algorithms can result in an considerable effectiveness improvement. This is the idea behind a new family of methods coined as *constrained clustering* [3], where the domain knowledge is introduced as rules in a generalised framework keeping the algorithm domain-independent. Two clear examples of this situation could be clustering data from multiple evidences of information, where introducing guiding data can be very useful, or in collections where the data has a very obvious grouping to which the traditional algorithm are biased, and where more interesting results could be found if we tell the algorithm to avoid that clustering.

These methods, called *"semi-supervised clustering"*, use background knowledge to impose some restrictions on the process, trying to influence the grouping that it finds in the data. This has been a very fruitful field in the last years [4,5,6,7,8,9,10,11,12]. This constrained clustering is quite different from a classification process, as the domain knowledge gives the clustering algorithm rules over data instances (documents), instead of examples of the categories. These rules reflect some preferences about whether or not the data instances should be in the same cluster, but it is still the algorithm which finds the groups in the data.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 322–329, 2009.

In this paper we propose a new framework of constrained clustering which, based on batch k-means, incorporates May and May-Not Link constraints as well as the Must and Cannot Link constraints proposed by Wagstaff et al. in [7], because in most real cases the domain knowledge is not categorical and only hints some traces or patterns. Thus, using absolute constraints could harm the algorithm effectiveness. Another contribution of this work is including unidirectional constraints, which could be interesting when working in certain domains.

After defining the new approach we tested it in an avoiding bias problem. In this real world clustering problem, the traditional algorithms tend to be biased to a dominant grouping, which is also well known, and the objective is to avoid that one, to discover new data interpretations. Our results in this experiment outperformed the Conditional Information Bottleneck-based method (CIB) [9], used as baseline. We also tested in other experiment the behaviour of the algorithm as the number of negative absolute and soft constraints is increased.

Next, in section 2 is presented the new framework. Section 3 describes the experiments and comments the results. Section 4 is devoted to the previous work about semi-supervised clustering, showing the differences with the proposed method. Finally, conclusions are reported in Sections 5.

2 K-means with Absolute and Soft Constraints

The k-means [13] algorithm is a very popular clustering method, due to its good trade-off between effectiveness and cost. It is a generic algorithm, which does not need any prior knowledge apart from the desired number of clusters. Moreover, its clear structure and flow makes extending and modifying it very easy.

In [7] Wagstaff et al. introduced in batch k-means two kinds of bidirectional instance level pairwise constraints, which were previously presented in [6]: *Must-Links*, connecting documents which must be in the same cluster and *Cannot-Links*, connecting documents which must not be in the same cluster. These constraints are absolute, i.e. a clustering has to fulfil all of them to be acceptable. While this absoluteness can be very convenient if we know categorically the relations between instances and we can not afford to have them misplaced, it could represent an excessive burden to the process. Indeed, as the authors admit in [7], it can lead to situations where, even though there is an acceptable solution, it can not be found as the outcome of the algorithm is extremely sensitive to the order in which the documents are inspected. For instance, it could be impossible to find a cluster for a document due to having a Cannot-Link constraint with a document in each cluster, a situation that might have not arisen if we had inspected the "conflictive" document earlier. Even when a solution can be found, the combination of absoluteness and sensitiveness to order can make the presence of constraints more detrimental than beneficial. For example, data instances connected with Must-Links will be dragged unconditionally to the cluster where the first of them is assigned, which could lead to worse clusterings.

In order to overcome these limitations we introduce in this paper two new kinds of soft (non-absolute) constraints, which will influence gradually the process instead of defining categorically where a document must or must not go:

May-links, connecting documents a and b if a is likely to be in the same cluster as b, and *May-Not-Links*, connecting documents a and b if a is not likely to be in the same cluster as b. These constraints are unidirectional, i.e, we are dealing with ordered pairs. In most domains the constraints will be reciprocal that is, (a, b) and (b, a) would be present. However, there could be others where this capability to express non-reciprocal constraints could be interesting. For instance, consider we want to cluster companies web-pages by industrial sector. It is sensible to assume that the pages of a company's products should be in the same cluster as their company main-page but not the opposite. This knowledge can be represented by a set of May-Links $(product_i, company_x)$. Another difference with the absolute constraints is that the May-Link and May-Not-Link constraints do not necessarily define a transitive relation.

The New Constrained k-means Algorithm. The resulting algorithm after introducing the absolute and soft constraints in the schema of the batch k-means is detailed in Fig. 1. The input data and parameters are: $\{x_1, \ldots, x_n\}$, the set of documents in the collection to cluster; k, the number of clusters that the algorithm will try to find; $musts$, $cannots$, $mays$ and $mayNots$, the background knowledge in form of constraints to be taken into account and w, the factor of influence of the soft constraints. The constraints $musts$, $cannots$, $mays$ and $mayNots$ are represented as sets of ordered pairs (in $musts$ and $cannots$ we will assume that a previous transitive closure has been taken and that, due their reciprocity, if (a, b) appears, (b, a) appears as well).

The first step (1) is initialising each cluster with a different document chosen randomly from the set of documents to cluster, as a sort of "iteration -1". Afterwards, and until the algorithm satisfies the convergence criterion (2) a loop is executed, where in each iteration the documents are assigned to a cluster using the function Assign, using the outcome of the previous iteration (*old*), the location of the documents already assigned in this iteration (*new*) and the previous set of clusters actualised by the changes made in this iteration (*current*).

Given a document x, the function Assign determines to which cluster it should be assigned. For each cluster j (3), the function tries first to honour the absolute constraints that affect x as in Wagstaff et al. [7] . That is, if x has a Must-Link with any of the documents already assigned to cluster j in this iteration (5), x is Put in that cluster and the function returns (6). Also, if there is a document with which x has a Cannot-Link (7), the cluster is discarded.

After testing the absolute constraints, the similarity of x with the centroid of the old cluster is calculated (10). This similarity value ($scores[j]$) will be modified by the soft constraints (13-16) affecting x. For each document which has been already assigned to this cluster in this iteration or has not yet been inspected and with which x has a May-Link, the *score* of the cluster j is increased in a certain amount w. If it has a May-Not-Link, the *score* of the cluster j is decreased in a certain amount w. This strategy fits well with the mechanism of the k-means algorithm, which uses information from an iteration (the centroid of the documents) in order to rearrange them in the next. Moreover, along with the non-absoluteness of the constraints, it lets the sole presence of these constraints affect gradually the clustering process while avoiding the problems exposed earlier. Once those steps have been tried on each cluster, the one with the highest *score* is chosen as the destination of x (20,21). If all the clusters were discarded the appropriate flag is returned (22), aborting the execution of the algorithm.

This new algorithm maintains the good computational behaviour of batch k-means: considering that k is the desired number of clusters, i the number of iterations, c is the number of constraints, and n the number of documents in the collection, our constrained k-means still is $O(k \times i \times n)$ in time. The searches in the constraint lists are not considered because compared with the document similarity calculation their cost is negligible. The algorithm is $O(k + n + c)$ in space, although again can be considered $O(k + n)$ because the space of storing the constrains is much smaller than the space for the documents.

```
CLUSTER($\{x_1, \ldots, x_n\}, k, musts, cannots, mays, mayNots, w$)
 1  $new \leftarrow$ SELECTRANDOMSEEDS($\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}, k$)
 2  while  convergence criterion has not been met
 3  do $current \leftarrow new$
 4      $old \leftarrow new$
 5      CLEAR($new$)
 6      for $i \leftarrow 1$ to $n$
 7      do
 8          $assigned \leftarrow$ ASSIGN($x_i, k, new, current, old, musts, cannots, mays, mayNots, w$)
 9          if $not(assigned)$
10             then error  "Impossible to cluster"
11      end
12  end
13  return $new$

ASSIGN($x, k, new, current, old, musts, cannots, mays, mayNots, w$)
 1  $scores \leftarrow [0, 0, \ldots, 0]$
 2  $assigned \leftarrow false$
 3  for $j \leftarrow 1$ to $k$
 4  do
 5      if $\exists x_i \in new[j]$ such that $(x, x_i) \in musts$
 6         then PUT($x, max, new, current, old$); return $true$
 7      if $\exists x_i \in new[j]$ such that $(x, x_i) \in cannots$
 8         then continue
 9      $assigned \leftarrow true$
10      $scores[j] \leftarrow$ SIMILARITY($x$, CENTROID($old[j]$))
11      for $h \leftarrow 1$ to $|current[j]|$
12      do
13          if $\exists(x, current[j][h]) \in mays$
14             then $scores[j] \leftarrow scores[j] + w$
15          if $\exists(x, current[j][h]) \in mayNots$
16             then $scores[j] \leftarrow scores[j] - w$
17      end
18  end
19  if $assigned$
20     then $max = indexof(\max(scores))$
21            PUT($x, max, new, current, old$); return $true$
22     else  return $false$

PUT($x, i, new, current, old$)
1  $current[clusterof(x, old)] \leftarrow current[clusterof(x, old)] \setminus \{x\}$
2  $current[i] \leftarrow current[i] \cup \{x\}$
3  $new[i] \leftarrow new[i] \cup \{x\}$
```

Fig. 1. k-means clustering algorithm with Must, Cannot, May and May-Not Links

3 Experiments and Results

The clustering algorithms try to detect an underlying organisation in the given data. Often, there is an obvious grouping of it, which is easily found by a simple manual examination. In that case, the clustering algorithms will be probably biased to fall in that organisation, which is not very helpful. The task of avoiding this grouping, trying to make the algorithm pay attention to other facts which could lead it to another unknown clustering, is called "Avoiding Bias", which, as well as having its intrinsic interest, will be used here to show the effectiveness of our constrained clustering algorithm. Besides, we also contribute a comparison of the behaviour of the Cannot and May-Not Links in a similar way as in [7].

In our experiments we have used two datasets used by Gondek and Hofmann in [9]: the first one (i) was created from WebKB's Universities dataset, taking only the documents from Cornell, Texas, Washington and Wisconsin universities and dropping those corresponding to "misc", "other" and "department" (1087 documents). The second one (ii) was created from Reuters RCV1 dataset, taking the documents with only one topic and region label and whose topic is MCAT or GCAT and whose region is UK or INDIA (1600 documents). As in [14], we have used as document representation the Mutual Information (MI) between a document and its terms. Cosine distance was used as similarity measure.

To compare the clustering yielded by the algorithm with a certain reference we have used three metrics [15], where higher values mean more similarity: Purity (P), a precision metric which measures how well the clustering results match the manual split in average, Mutual Information (MI), a metric which measures how much information about a clustering is conveyed by another and Rand Index (RI), which measures the ratio of good decisions made by the algorithm.

Experiment 1: Avoiding Bias. In this experiment we have used the datasets defined above in order to address an Avoiding Bias problem. Each document is categorised according to two different criteria, so we will take one of these criteria as the known clustering of the data and we will try to avoid it, using the constrained k-means algorithm that we have introduced. After the algorithm is executed, we will measure the similarity of the final set of clusters with the known clustering and with the other one present in the data.

The constraints set is created with two May-Not-Link constraints for each pair of documents (i.e. both directions) belonging to the same cluster in the clustering we are trying to avoid (which is already known for us). These are the only constraints that are going to be used in the clustering process. Specifically, the Cannot-Link constraints are unsuitable for this task due to their absoluteness.

In order to produce a fair comparison between algorithms, we have set in each run k to the number of groups of the expected (i.e., non avoided) clustering. To tune w (the weight of the soft constraints) we have used a crossvalidation strategy, which involved testing the possible values in dataset (i) and taking the one with best results ($w = 0.0025$), using that value in the other dataset. Also, the convergence condition is tested comparing the centroids of the present iteration with those of the previous one. The process is stopped as well if a certain number of iterations is exceeded without convergence.

In Table 1 we show the results achieved by CIB, our algorithm and a batch k-means in this experiment. As in the last two algorithms the outcome of the clustering process is very dependant on the initial seeds the results shown are the average of 10 random seed initialisations. In each of these initialisations we have as well randomised the order in which the documents were inspected.

As a previous note we should stress how the MI values of the runs of the batch k-means in the datasets show unequivocally the tendency of that algorithm to one of the possible clusterings of the data, showing a real-world example where having a way to avoid that bias could come in handy.

With the trained w our algorithm performed really well, achieving the two aims of the Avoiding Bias task. Firstly, we have been able to avoid the known organisation of the data, which is visible in the considerable decrease of the values of MI for the known clustering of our algorithm and batch k-means. Secondly, the outcome of our clustering algorithm resembles more the not known organisation of the data than the known one, which can be confirmed comparing the MI for the known and unknown clustering. Moreover, in all cases the quality of the clustering, measured by the Purity for the not known clusterisation, is still high.

Comparing with the results of Gondek and Hoffman (CIB), our algorithm achieves in almost all cases noticeable increases in the similarity to the unknown clustering than their approach, with also more quality. The only exception happens in dataset (ii) when trying to avoid the "Region" criterion. This can attributed to the special nature of this dataset, which is extremely unbalanced. Nevertheless, we must stress that even in this extreme case the algorithm is able to fulfil the two aims previously pointed out.

Experiment 2: Incremental Behaviour. We have used dataset (i) to compare the behaviour of the soft and absolute negative constraints as their number is increased. Now we are not trying to avoid any clustering, but to achieve the maximum similarity (measured with RI) with the ground truth (the University criterion). The constraints were defined over nine tenths of the documents, taking randomly pairs of documents belonging to different clusters. We used this crossvalidation strategy, similar to the one used in [7], to see the direct influence

Table 1. Results for the avoiding bias experiment with the defined datasets for batch k-means, the new constrained k-means working with soft constraints (SCKM) and the CIB based method

Dataset (i)	Avoiding Topic (k=4)			Avoiding University (k=5)		
	MI(Topic)	MI(Univ.)	P(Univ.)	MI(Univ.)	MI(Topic)	P(Topic)
CIB	0.0067	0.0189	0.2917	0.0085	0.2342	0.4735
Batch k-means	0.5177	0.2111	0.4395	0.3217	0.5164	0.6730
SCKM (w=0.0025)	0.0039	0.2947	0.5061	0.0031	0.4686	0.6431

Dataset (ii)	Avoiding Topic (k=2)			Avoiding Region (k=2)		
	MI(Topic)	MI(Region)	P(Region)	MI(Region)	MI(Topic)	P(Topic)
CIB	0.0015	0.0107	0.5516	0.0001	0.8548	0.9781
Batch k-means	0.0073	0.0814	0.8253	0.0965	0.0081	0.9838
SCKM (w=0.0025)	0.0003	0.1408	0.8253	0.0004	0.0054	0.9838

of the constraints on the whole collection and the indirect influence over the non constrained documents. The results showed that, although with few constraints (< 2000) the behaviour of absolute and soft constraints is similar, improving slightly the results of batch k-means, increasing the number of absolute constraints entails a decrease of the effectiveness, well below batch k-means, a situation which does not arise with the soft constraints, which experiment a linear improvement with the number of constraints. So it has been demonstrated that in this kind of problems the soft constraints outperform the absolute constraints, which are not adequate when working with more than a few constraints.

4 Related Work

The way in which the soft constraints are introduced in our algorithm is similar to the one presented in [4] by Yang and Callan. However, they use the constraints in an algorithm specially tailored for the task of near duplicate detection. Also the algorithm only used the Must, Cannot and "Family" (similar to May) rules and they are only bidirectional. Another key difference is that their algorithm does not take advantage of the information from the previous iteration.

Also in the field of IR Ji and Xu presented in [5] a semi-supervised clustering method based on spectral clustering that is very effective, but only allows the inclusion of background knowledge through soft pairwise relations of membership to the same cluster. The method is quite time consuming, as it implies the calculus of the eigenvectors of the document matrix. In [8] Klein et al. present a constrained hierarchical clustering including Must and Cannot Links. The algorithm has the problem of the computational cost of the hierarchical methods but it outperforms the Wagstaff et al. method in terms of effectiveness. However, they only evaluated it in synthetic and very small non-textual collections.

Several papers were presented recently in DM forums; one of them was the mentioned seminal paper in finding alternative clustering presented by Gondek and Hofmann [9]. They introduced an approach that uses the Conditional Information Bottleneck theory using a dual objective function searching for both alternative and good clustering. One problem of this technique is that it requires a joint distribution information for each variable and that is not always available. In [10] Bae and Bailey presented a constrained clustering method, enabling the Cannot-Link rules, based on a average-link algorithm. Although it outperformed CIB, the algorithm complexity makes it inefficient for large collections.

Some papers approach the inclusion of the constraints through the learning of distance functions [16], such as Davidson and Qi [11], which uses Must-Link and Cannot-Link knowledge but implies the use of Singular Value Decomposition (SVD), or Cui et al. [12], an approach to produce multiple orthogonal clustering views using Principal Component Analysis (PCA).

5 Conclusions

In this paper we have presented a general algorithm for constrained clustering extending the well-known constrained k-means [7] with soft-constraints. With this inclusion we still have a clustering algorithm with high performance and able to work with large text collections. The new soft-constraints allow tackling the task of avoiding bias and outperform the CIB-based method [9], specially designed for that task. Our algorithm also presents a good behaviour when the number of constraints is reduced, sharing this property with other algorithms more expensive computationally like the CCL [8], and it does not degrade the effectiveness when increasing the amount of constraints but the opposite.

Acknowledgements. This work was co-funded by FEDER, Ministerio de Ciencia e Innovación, Xunta de Galicia and Ministerio de Educación under projects TIN2008-06566-C04-04 and 07SIN005206PR and FPU grant AP2007-02476.

References

1. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. ACM Computing Surveys 31(3), 264–323 (1999)
2. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34(1), 1–47 (2002)
3. Basu, S., Davidson, I., Wagstaff, K.: Constrained Clustering: Advances in Algorithms, Theory, and Applications. Chapman & Hall/CRC, Boca Raton (2008)
4. Yang, H., Callan, J.: Near-duplicate detection by instance-level constrained clustering. In: Proc. of SIGIR 2006, pp. 421–428 (2006)
5. Ji, X., Xu, W.: Document clustering with prior knowledge. In: Proc. of SIGIR 2006, pp. 405–412 (2006)
6. Wagstaff, K., Cardie, C.: Clustering with instance-level constraints. In: Proc. of ICML 2000, pp. 1103–1110 (2000)
7. Wagstaff, K., Cardie, C., Rogers, S., Schrödl, S.: Constrained k-means clustering with background knowledge. In: Proc. of ICML 2001, pp. 577–584 (2001)
8. Klein, D., Kamvar, S.D., Manning, C.D.: From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In: Proc. of ICML 2002, pp. 307–314 (2002)
9. Gondek, D., Hofmann, T.: Non-redundant data clustering. In: Proc. of ICDM 2004, pp. 75–82 (2004)
10. Bae, E., Bailey, J.: COALA: A novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In: Proc. of ICDM 2006, pp. 53–62 (2006)
11. Davidson, I., Qi, Z.: Finding alternative clustering using constraints. In: Proc. of ICDM 2008, pp. 773–778 (2008)
12. Cui, Y., Fern, X.Z., Dy, J.G.: Non-redundant multi-view clustering via orthogonalization. In: Proc. of ICDM 2007, pp. 133–142 (2007)
13. McQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proc. of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297 (1967)
14. Pantel, P., Lin, D.: Document clustering with committees. In: Proc. of SIGIR 2002, pp. 199–206 (2002)
15. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
16. Xing, E.P., Ng, A.Y., Jordan, M.I., Russell, S.: Distance metric learning, with application to clustering with side-information. In: Advances in Neural Information Processing Systems, vol. 15, pp. 505–512. MIT Press, Cambridge (2003)

Optimizing WebPage Interest

Willem Elbers and Theo van der Weide

Institute for Computing and Information Sciences,
Radboud University, Nijmegen,
The Netherlands
ru@willemelbers.nl, tvdw@cs.ru.nl

Abstract. In the rapidly evolving and growing environment of the internet, web site owners aim to maximize interest for their web site. In this article we propose a model, which combines the static structure of the internet with activity based data, to compute an interest based ranking. This ranking can be used to gain more insight into the flow of users over the internet, optimize the position of a web site and improve strategic decisions and investments. The model consists of a static centrality based component and a dynamic activity based component. The components are used to create a Markov Model in order to compute a ranking.

Keywords: web graph; interest; centrality; user flow; Markov Model.

1 Introduction

Users are entering the world wide web by accessing a web site and use the available hyperlinks to travel to other pages and web sites. Simultaneously web site owners are constantly updating their existing web sites and creating new web sites. Over time web sites might also cease to exist. In short, users follow the structure created by web masters and others while this structure is constantly evolving. In this article we want to investigate how we can gain more insight into the static structure of the internet and the dynamic flow of users through this structure. This results in a flow potential score for a web site. The improved insight, based on the flow potential score, can result in more strategic decisions and investments.

Flow potential, which is more than just flow if it also depends on properties of the underlying structure, will be referred to as web site interest. The research question in this article is: How can web site interest be measured based on static and dynamic properties? In order to answer this question, the following sub questions have to be answered: (1) What are the static and dynamic properties of web sites?, (2) How can these properties be measured? and (3) How can these two types of properties be combined?

In section 2 of this article the model is introduced. The static and dynamic properties will be specified in the context of an experiment, discussed in section 3. The initial results are presented in section 4 and section 5 will conclude this article.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 330–337, 2009.

2 The Model

The model, proposed in this section, derives a web page interest value $R(p)$ of a web page p from the following two components. The first component is the web page importance $S(p)$, which is measured relative to other web pages. The second component $D(p)$, is a property that quantifies the interest in page p. These components are combined by a function called R_c:

$$R(p) = R_c(S(p), D(p)) \, . \tag{1}$$

The two components combine static and dynamic properties of web pages respectively. The importance function S is a static property of the (web)graph and may be measured by centrality, which is a known concept from graph theory. Centrality is a measure to indicate the importance of a node in the graph, based only on the structure of the graph. The four most known centrality measures, introduced by Freeman [1] and Bonacich [2], are degree centrality, betweenness centrality, closeness centrality and eigenvector centrality. These centrality measures also have a conceptual meaning. The degree centrality measures the potential of a node to be part of the flow in a graph. Betweenness centrality can be seen as the potential of a node to control the flow in a graph. Closeness centrality can be seen as the potential of a node to avoid the control potential of other nodes in a graph. Eigenvector centrality is a measure for how connected to other influential nodes a node is in a graph. Besides these centrality measures, there are also two well known algorithms which use the static web graph to rank pages: PageRank [3] and HITS [4].

The interest function D is a flexible, dynamic, component. Link traversal counts how often users follow specific links. This would be the best activity based measure in the case of website interest. Unfortunately this information is, usually, not publicly available. Even the number of visitors of a web site is hard to obtain. We will propose a solution to convert activity based data for nodes into probabilities of following a link.

So far, the components in the model have been introduced, their relation has not. The solution is based on work in the field of adaptive web sites, [5] [6] [7] and especially [8] [9] [10] [11]. Using a Markov Model seems to be a promising solution for R_c. The m nodes of a graph are the states of the Markov Model. The, structural, centrality measure can be used to create an $1 \times m$ initial probability distribution, L, and the activity based data, which are transformed into transition probabilities, can be used as the $m \times m$ one step transition probability matrix Q. Then R will be the $1 \times m$ ranking vector $R = L \times Q^k$, based on taking k steps through the graph. At some point, for a large enough k, a steady state is reached where increasing k further has no effect anymore. That state is also independent of the initial probability distribution and at that point the ranking will only be activity based.

In the remainder of this article we will use the following definitions for graphs. A (web)graph G is defined as an ordered pair $G = (V, A)$ where $V = \{p_1, \ldots, p_n\}$ is the set of vertices or nodes and A is the set of arcs between the nodes in the graph, defined as the set of ordered pairs $(v, w) \in A \subseteq V^2$. We can also write $v \rightarrow w$ or $A(v, w)$ to specify an arc in the graph. In the case of a web graph, the nodes of the graph are the actual web pages and the arcs are the actual hyperlinks between these web pages.

3 The Model into Action

3.1 The Static Property

Based on the static graph structure we have to compute a centrality score for each node. Four methods to compute centrality have been mentioned in section 2. Based on their conceptual meaning, betweenness centrality seems like a very promising candidate. This is a measure for the potential of a node in the graph to control the flow. If many people pass through a site, z, when following links from site v to site w, then this z has a high potential to control where those people are going.

In order to optimize the calculation for betweenness centrality, we will use ego betweenness, introduced by Everett and Borgatti [12]. Ego betweenness of a node v is the betweenness score of that node in its ego network as defined by Freeman [13]. The ego network of a node is the graph with the node itself, all the direct neighbors of this node and the arcs between these nodes in the original graph. The betweenness for each node is needed, therefore n ego networks have to be computed. The advantage of these ego networks is that they will be relatively small. We have approximately 12K, uniquely connected, nodes in the test dataset, but the average ego network size is only 10 nodes and the biggest ego network is around 250 nodes. How do we extract the ego network, $G_{ego} = (V_{ego}, A_{ego})$, for a node $v \in V$ from graph $G = (V, A)$? Based on this definition, two properties hold: (1) $V_{ego} \subseteq V$ and (2) $A_{ego} \subseteq A$ and based on the definition of Freeman all direct neighbors and their arcs of v need to be included.

We perform two steps to extract the ego network. First we will get the set with all nodes in the ego network for a certain node v: $V_{ego} = \{v\} \cup \{w \in V | (v, w) \in A \vee (w, v) \in A\}$. Second, based on the set with nodes in the ego network, V_{ego}, we can construct the set of arcs in the ego network, A_{ego}. If an arc $(v, w) \in A_{ego}$, exists in A then it should also exist in A_{ego}: $A_{ego} = \{(v, w) \in A | v \in V_{ego} \wedge w \in V_{ego}\}$.

Now that we have the ego network for a node v in place, its actual ego betweenness score, c_b, can be computed. Let $B_{ego} = A^2_{ego} \times (1 - A_{ego})$ where 1 is a matrix with only ones of the same dimension as A_{ego} and × is the cell-wise multiplication operator for matrices. The ego betweenness is the sum of the reciprocals for the non zero entries in B_{ego}:

$$c_b = 1 / \parallel B_{ego} \parallel_1 \ . \tag{2}$$

If the ego betweenness is computed for all nodes in the graph, the result will be a vector C_b with these scores for each node. Next, this vector is transformed into the initial probability distribution by dividing each centrality score by the sum of all centrality scores:

$$I = 1 / \parallel C_b \parallel_1 \times C_b \, . \tag{3}$$

Degree centrality could also be an interesting measure to use. It is the potential of being part of a flow in the graph. However, if you are not part of any shortest paths between two web sites, people are more likely to follow the shorter paths and not enter your web site. Degree centrality is an easy to compute centrality measure, therefore it might be interesting to compare degree centrality based rankings to betweenness centrality based rankings.

Closeness centrality is the potential of a node to avoid being part of the flow. Since we are interested in optimizing the flow to our own web site, we are not so much interested in web sites which can avoid the flow of other web sites. This could be a desirable measure if information independence is very important.

Eigenvector centrality is a measure which increases a nodes importance if it is connected to other important nodes. Since this centrality measure is less aimed at how a node can influence the flow in a graph, we didn't choose to use this centrality measure. However, after the first experiments, it could be interesting to see how this centrality measure fits in and performs.

PageRank would also be a interesting measure to use for the static property. To put it simple, a high PageRank is an indication for the number of incoming pages and their PageRank. Therefore it seems quite likely to say there should be some relation between a high PageRank and a high flow, however many incoming links do not necessarily mean a lot of incoming traffic. We think this is the most interesting alternative to look into for any future research. The HITS algorithm assigns hub and authority scores to the nodes in the graph. It is not obvious how this relates to the flow in a graph, since the number of links doesn't say anything about traffic numbers directly. The chance on more traffic might be bigger with more incoming or outgoing links, but this requires further research.

3.2 The Dynamic Property

As mentioned already, it would be ideal to have link traversal or traffic data of all web sites on the internet. Unfortunately this is not possible. We have come up with a different approach to work around this problem. This approach is applicable to websites as well as blogs, as long as usage data is available for the node in the graph. Because we have a dataset of the dutch blogosphere, we have come up with a solution based on timestamps as a measure for blog activity. This method can also be used for fora, but for websites a different approach has to be used in order to retrieve the activity based data. The basic concept of converting node based activity into link traversal activity, as proposed in the following sections, is also applicable to websites as a whole instead of blogs only.

Since the dataset contains blogs, we will propose a method to crawl activity based data from blogs. Blogs often have the option to post reactions with a topic. These reactions can be characterized by a time stamp on the page. For our experiment we will gather all time stamps associated with a blog and use this as the activity measure for the dynamic property. This approach is based on the assumption that reactions to a blog are related with the traffic of that blog. This approach has a big advantage, it's easy to add into the crawling process which analyzes the blogs to construct the graph structure. By using these time stamps as a measure for the number of reactions on a blog, we have an easy way to obtain a measure for the activity on a blog.

Let $G = (V, A)$ be a graph consisting of a set of nodes V and a set of arcs, $A \subseteq V^2$. For each node $v \in V$ the function $r(v)$ returns the activity measure for the supplied node v. In our case this activity measure is the number of reactions that were posted on a blog. Blog visitors are traveling this network structure. Let $P(w|v)$ be the probability the visitor follows a link to node w given the fact he is currently in node v. These probabilities have to be estimated from the activity measure. So basically we are constructing a flow network, where each link has an unbounded capacity.

Besides by following links, the activity measure of a blog will originate from visitors starting in a particular node. Visitors may also stop in certain blogs. This is modeled by adding two nodes *source* and *sink* to the graph and create arcs from *source* into each blog and also links from each blog to *sink*. The resulting graph is denoted as $G' = (V', A')$. The activity measure of the new nodes still needs to be defined. Of course, $r(\mathit{source})$ is the number of unique visitor to the blog graph. Obviously $r(\mathit{source}) = r(\mathit{sink})$.

We assume the flow through a link (v, w), from node v to node w, amounts to: $P(w|v) = r(v)$. When traversing a link, we assume it is more likely to take a link to a node with a higher activity measure. In other words: $P(w|v) \geq P(z|v)$ if and only if $r(w) \geq r(z)$. Based on this assumption $P(w|v)$ is defined as follows:

$$P(w|v) = \begin{cases} d(v)\frac{r(w)}{R(v)} & \text{if } v \longrightarrow w \in A \\ 1 - d(v) & \text{if } w = \mathit{sink} \\ 0 & \text{if } w = \mathit{source}\,. \end{cases} \tag{4}$$

where

$$R(v) = \sum_{w \neq \mathit{sink} \in V' : v \rightarrow w} r(w)\,. \tag{5}$$

and $d(v)$ is a damping factor that determines the likelihood a visitor stops in a particular blog. It should hold that the sum of all probabilities equals to one, this is shown in the following proof:

$$\begin{aligned} \sum_{w \in V' : v \rightarrow w} P(w|v) &= \sum_{w \in V : v \rightarrow w} P(w|v) + P(\mathit{sink}|v) + P(\mathit{source}|v) \\ &= \sum_{w \in V : v \rightarrow w} d(v)\frac{r(w)}{R(v)} + (1 - d(v)) \\ &= \frac{d(v)}{R(v)} \sum_{w \neq \mathit{sink} \in V' : v \rightarrow w} r(w) + (1 - d(v)) \\ &= d(v) + 1 - d(v) \\ &= 1 \end{aligned}$$

The flow conservation law states that incoming flow and outgoing flow, of a node v, should be equal. This should also hold for this model:

$$\sum_{w \neq sink \in V' : w \rightarrow v} P(v|w)r(w) = r(v) = \sum_{w \neq source \in V' : v \rightarrow w} P(w|v)r(v)\,. \qquad (6)$$

If we look at the incoming flow we can derive some properties by looking at the following cases:
1) if $v \in V$, we conclude:

$$\frac{1}{d(v)} = \sum_{w \neq sink \in V' : w \rightarrow v} \frac{r(w)}{R(w)}\,. \qquad (7)$$

2) if $v = sink$, we conclude:

$$r(sink) = r(source) = \sum_{w \in V : w \rightarrow sink} (1 - d(w))r(w)\,. \qquad (8)$$

3) obviously, if $v = source$ the sum results in 0.

3.3 The Algorithm

Looking back at what we have discussed so far, we have made the following choices in the context of the proposed experiment.

1. the static property, $S(p)$, will be the initial probability distribution based on ego betweenness centrality: $I = 1/ \parallel C_b \parallel_1 \times C_b$.
2. the dynamic property, $D(p)$, will be the one-step transition matrix based on the reactions (time stamps) found on the blog p.
3. The relation is defined by the Markov Model $< S, Q, L >$ where S is the set of blogs (nodes), Q is the one-step transition matrix and L is the initial probability distribution.

Based on these choices we have developed an algorithm to compute rankings on our data set. The algorithm can be divided in several steps, the first three steps are the initialization steps and the fourth step is the actual ranking computation. This is a very basic description of the steps needed in the algorithm, no optimizations have been applied.

1. Construct the one-step probability matrix from the weighted graph.
2. Compute the ego betweenness for all nodes, based on the static structure.
3. Compute the initial probability distribution from the ego betweenness values.
4. Compute the rankings for all nodes based on a history of m steps.

4 Initial Results

In this section we will very briefly cover the results we have seen so far. Based on the ideas presented in this paper we are developing a prototype. The prototype is for the most part implemented as described in this article. The current prototype can use a betweenness and (in)degree centrality measure. If we look at the indegree based initial probability distribution we see almost similar results to the ranking created by the supplier[1] of the data set. Since they also use an indegree based ranking, this should be true.

If we use the betweenness bases approach with a constant value for d, the results look promising but they have also brought a problem with the dataset to our attention. We haven't been supplied with activity data for all blogs. The data is available, therefore we expect the result to improve even further if we run the algorithm on the correct dataset and by running the algorithm with a proper implementation of the value for d.

5 Conclusion and Future Work

In order to answer the research questions, a model has been presented and it has been discussed in depth in the context of an experiment. The importance of a web site, the static component, can be measured by using any of the known centrality measures. Based on their conceptual meaning, we have chosen to primarily use betweenness centrality. In order to optimize the algorithm we have implemented an ego betweenness algorithm. The dynamic property has been defined as the number of reactions to the postings of a blog. And an approach to convert node activity into link activity has been proposed. This approach is also applicable to websites as a whole. In the most ideal situation however, we should have access to traversal information between web sites. The relation between the static and dynamic property is defined by a Markov Model, inspired by research conducted into the field of adaptive web sites. The dynamic property is used to construct the one-step probability transition matrix, Q, the static property is used to compute the initial probability distribution, L, and the states, S of the Markov Model are the unique blogs, the nodes in the graph. In order to optimize interest to a given web site, the Markov Model is used to compute a ranking based on a depth of m navigational steps. By creating links, advertising for example, to the highest ranking web sites, we can optimize interest for the given web site.

Obtaining this dynamic information is a problem. Traffic data is not freely available. We propose a solution for this problem in the domain of blogs (and possibly other community based areas). Instead of traffic we will measure reactions to a posting. This has two disadvantages. (1) The results might be polluted with 'wrong' reactions. This can be solved by improving the crawling algorithm. (2) The other disadvantage is the fact we actually need transition or traversal numbers. If we measure reactions, it's a activity measure of a node in the graph, not an arc. In order to translate the node activity numbers to traversal numbers, we made the assumption it is more likely for people to leave for a page with more visitors. Based on this assumption an approach has been presented to compute transition weights.

[1] SiteData B.V. (www.sitedata.nl)

5.1 Future Work

Based on the foundations presented so far, some topics are still open and others raised more questions.

1. Perform the described experiment.
2. Incorporate other centrality measures for the $S(\mathrm{p})$, especially PR.
3. Extend the measuring of activity, both for blogs and for websites.
4. Research solutions to get actual traffic and/or traversal information.

References

1. Freeman, L.C.: Centrality in social networks - conceptual clarification. Social Networks 1(3), 215–239 (1979)
2. Bonacich, P.B.: Factoring and weighing approaches to status scores and clique identification. Journal of Mathematical Sociology (2), 113–120 (1972)
3. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web (1999)
4. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment (1999)
5. Perkowitz, M., Etzioni, O.: Adaptive sites: Automatically learning from user access patterns. Technical report, Department of Computer Science and Engineering. University of Washington, Seattle (1997)
6. Garofalakis, J., Kappos, P., Mourloukos, D.: Web site optimization using page popularity. Technical report, University of Patras, Greece (1999)
7. Zhou, B., Chen, J., Shi, J., Zhang, H., Wu, Q.: Website link structure evaluation and improvement based on user visiting patterns. In: HYPERTEXT 2001: Proceedings of the twelfth ACM conference on Hypertext and Hypermedia, pp. 241–244. ACM Press, New York (2001)
8. Sarukkai, R.R.: Link prediction and path analysis using markov chains. In: Proceedings of the 9th international World Wide Web conference on Computer networks: the international journal of computer and telecommunications netowrking, pp. 377–386. North-Holland Publishing Co., Amsterdam (2000)
9. Zhu, J., Hong, J., Hughes, J.G.: Using markov models for web site link prediction. Technical report, School of Information and Software Engineering. University of Ulster at Jordanstown (2002)
10. Zhu, J., Hong, J., Hughes, J.: Using markov chains for link prediction in adaptive web sites. In: Proc. of ACM SIGWEB Hypertext, pp. 60–73. Springer, Heidelberg (2002)
11. Eirinaki, M., Vazirgiannis, M., Kapogiannis, D.: Web path recommendations based on page ranking and markov models. In: WIDM 2005: Proceedings of the 7th annual ACM international workshop on Web information and data management, pp. 2–9. ACM Press, New York (2005)
12. Everett, M., Borgatti, S.P.: Ego network betweenness. Social Networks 27(1), 31–38 (2005)
13. Freeman, L.C.: Centered graphs and the structure of ego networks. Mathematical Social Sciences 3(3), 291–304 (1982)

The “Beautiful” in Information
Philosophy of Aesthetics and Information Visualization

Gerald Benoit

Dept of Computer Science & GSLIS, Simmons College, 300 The Fenway, Boston, MA
benoit@fas.harvard.edu, benoit@simmons.edu

Abstract. At the intersection of retrieval and visualization are opportunities to learn more about IR’s view of knowledge and evidence by considering Kantian and post-modern philosophies of aesthetics[1].

1 Aesthetics

The philosophy of aesthetics refers not to the popular idea of graphic design [1] but to the relationship of experience and truthful sense of reality. What is beautiful, then, is something that bears a relationship with truth. “Beauty” must be set out as a proposition “that p”; metaphysical truth viewed here as either the Enlightenment’s independent, objective reality to be revealed or post-modern (PoMo) relativist, subjectivist, contextualized “warranted assertions.” In IR, retrieval sets are ranked by the system designers according to their models of human language behavior [2]; the retrieval set’s members become a proposition, “that p,” that awaits the end-users relevancy judgment to provide the context in which the proposition finds the necessary condition to make it true. Introducing information visualization (IV) as a way of using abstract images to help users establish significance both adds a layer of complexity and a philosophically-based path to viewing IR differently. In this poster, a reduction from a larger detailed analysis, two streams of philosophical thought are considered - Kant’s aesthetics and post-modernism - to suggest that IR modelers are responsible agents in the interpretation of IR sets, that aesthetics lead to different ways of understanding IR and the user, and that there is at the core of IV/IR a seeming conflict between *a priori* forms of knowledge and *a posteriori* relativism. While truth and beauty are foreign ideas to IR, in this poster, we consider the concept of information as a function of the proposition of “beauty.”

1.1 Kant

It is usually given that end-users provide the determination of relevancy of the query to the document collection representation from a combination of their experiences and needs with an IR system’s relevancy ranking. However the process of establishing significance is not necessarily the traditional sense/referent pair

[1] The full discussion is available at http://web.simmons.edu/˜ benoit/ICTIR09.pdf).

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 338–341, 2009.

or a question of naming [3][4][5][6]. "Information" is the result of data which purpose has been established, something to aid end-users in their lifeworld, for which the user can provide a warrant rather than "because the system said" [7]. From Kant's perspective [8], a proposition or fact that is "beautiful" asserts a claim on *a priori* knowledge. Regaining this knowledge requires dividing the sense of an object into the viewer's/user's *sensibility* and *understanding* of the object as the user tries to establish an object's meaningful *purpose*, sparked by interpreting the visual object. To understand an object, or to be aesthetically informative, there needs to be a basis that justifies why others would find it also beautiful, a claim to subjective interpretation as well as a claim to universality, otherwise the object is reduced to merely "pleasing." Kant [8] held that the relationship between reality and sense must be without undue external influence, *ohne alles Interesse*. End-users first *experience* beauty and then *apprehend* its form - two distinct activities - when establishing an object's definitive *purpose*, the way a knife, for instance, has a form that makes sense because we understand what it is supposed to be. This challenges how to create visualizations of non-real entities, abstractions in the retrieval interface that stimulate apprehending purpose (cf. [9]). Yet that people do accept visualizations suggests there must be some necessary condition that is satisfied [§12] to legitimate judgment and imputes the same satisfaction necessary from all. A "double-reflexive" reciprocal relationship is necessary between designers, the viewer's self-reflection, and the larger society, to create the "purposive state of mind" [§40] necessary to transform data to a warranted basis for further action by the user in his lifeworld [10], [11].

1.2 Post-Modernist Ideas

The postmodern perspective, especially expressed by Baudrillard [12], Levis [13], Fielder and Jameson [14] reconsider what reality is and where it is in relation to its supposed reproduction. The PoMo stance questions whether *a priori* reality exists at all and is subservient to representations or whether what is graphically presented is only a "simulation" of reality. The PoMo view is itself contradictory, arguing across a spectrum from favoring an élite of aesthetic experts who guard against trivializing, uncritical and facile production to hyper-relativism, to "promote that which is flexible, pluralistic, and hospitable to the popular [we] are no longer able to invent new styles and worlds the genius is in the blend, not in breathtaking innovation" [14]. Consequently the usual basis of establishing truth is lost in favor of relevancy judgments and knowledge warranted solely by instrumentalism. Baudrillard predicted this and anticipated our obsession with images as "pre-given reality," without any way to measure truthhood or falsity. Indeed the whole PoMo program can be summed up by Heim writing about digital objects: "they do not 're-present' a real thing" [15]. Thus only what coheres in the user's mind, even if that means true belief in false facts, is reality. This seeming illogic exposes what some positivists discovered [16] and so lost their way: that at the root of IR is a continuation of the Kantian, although unacknowledged, belief that universals exist and that we expose them; yet "relevancy" and truth are conditioned on socially constructed facts. These impacts IR because the expectation of relevancy judgments by the individual user favors a relativist, *a posteriori* view of knowledge while the algorithms of IR models are unacknowledged functions of a Kantian *a priori* reality and truth. Both paths - empiricist and postmodern - ultimately rely on the transcendent.

2 Dilemma

To the strong empiricist program, IR as a science cannot admit of the transcendental and rejects it as a legitimate source of evidence. An aesthetic turn might aid sincere inquiry into the necessary presuppositions of knowledge that are applied in IR algorithms and consequently expand the scope and objective validity of knowledge and IR system design. From PoMo we might doubt what we can know and how we can know. The expanding use of IV in IR is an opportunity to move from prima facie measurements of IR effectiveness to learn more about the user's experience, critique the IR modelers' role, and establish the legitimacy of visual-intensive technologies. While these ideas are often dismissed by some derisively as "it's all relevancy" or "it's the user who determines relevancy", holding that the ides of truth and beauty do not attach to technology, IV/IR expressed through aesthetics provides a shareable framework for exploring visual objects role in the end-user experience and translates into experimental forms acceptable to IR [10]. Echoed through PoMo, it stimulates critique of IR modelers' undue influence on relevancy judgments and the privileging of certain forms of evidence. Finally, aesthetics asks whether IR as a discipline recognizes its *a priori* claims of knowledge and its contradictory stance towards social constructivism. In this poster, it is argued that a middle way, shaping the Kantian model to see how aesthetics of beauty and purpose contribute to empirical research, but sensitive to modern individualism. There is a recognized gap between the two that a consideration of the "beauty of information" applied to IR exposes and which it might fill.

References

1. Chen, C.: Top 10 unsolved information visualization problems. IEEE 25(4), 12–16 (2005)
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern information retrieval. ACM Press, New York (1998)
3. Frege, G.: On sense and reference. In: Translations from the Philosophical Writings of Gottlieb Frege. Blackwell, Oxford (1960)
4. Kripke, S.: Naming and necessity. Harvard University Press, Cambridge (1980)
5. Russell, B.: On denotating. In: Marsh, R.C. (ed.) Logic and Knowledge, George Allen and Unwin, London (1956) (reprinted in)
6. Russell, B.: Knowledge by acquaintance and knowledge by description. In: Mysticism and Logic. Doubleday, Garden City (1957)
7. Ulhířová, L.: On the role of the PC as a relevant object in face-to-face communication. J. Pragmatics 22(5), 511–527 (1994)

8. Kant, I.: Kritik der Urteilskraft (Critique of judgment). Hacket, Indianapolis (1790/1987)
9. Herr, J.: Voyagers and voyeurs: supporting social data analysis. CIDR (2009), http://www-db-cs-wisc-edu/cidr/cidr2009/JeffHerrCIDRKeynote.pdf
10. Benoit, G.: Information seeking as communicative action. Ph.D. Diss, UCLA (1998)
11. Habermas, J.: Erkenntis und Interesse. Mit einem neuen Nachwort. 2. Aufl. Suhrkamp, Frankfurt am Main (1973)
12. Baudrillard, J.: System of objects. Verso, London (1968)
13. Levis, F.R.: For continuity. Cambridge Univ. Press, Cambridge (1933)
14. Jameson, F.: Cross that border - close that gap: postmodernism. In: Reprinted in The Collected Essays of Leslie Fiedler, vol. 2 (1971)
15. Heim, M.: Cyberspace/Cyberbodies/Cyberpunk. Sage, London (1975)
16. Coffa, J.A.: The semantic tradition from Kant to Carnap and the positivists of the Vienna Circle. Cambridge Univ. Press, Cambridge (2002)

IR Evaluation without a Common Set of Topics

Matteo Cattelan and Stefano Mizzaro

Dept. of Mathematics and Computer Science
University of Udine
Via delle Scienze, 206
33100 Udine, Italy
matteo.cattelan@gmail.com, mizzaro@dimi.uniud.it

Abstract. Usually, system effectiveness evaluation in a TREC-like environment is performed on a common set of topics. We show that even when using different topics for different systems, a reliable evaluation can be obtained, and that reliability increases by using appropriate topic selection strategies and metric normalizations.

Keywords: IR effectiveness, TREC, topics.

1 Introduction

We can model TREC-like experiments as in Tab. 1: effectiveness of each run r_i on each topic t_j is evaluated (usually by means of Average Precision, AP, although other metrics can be, and indeed are, used). AP values are then averaged to obtain an overall measure for each run. Usually the arithmetic mean is used, and Mean AP (MAP) is obtained. Alternatives do exist also here, e.g., GMAP, that uses the geometric mean and is used, e.g., in the Robust track. The obtained measure can then be used to rank runs/systems according to their effectiveness.

Although the TREC-like evaluation paradigm is quite stable, research to improve it and to make it more effective has been going on in the last decade. The number of topics is an issue that has undergone major attention [8,11,3,9,7]. Recently it has been shown that, at least in principle, fewer topics could be used, provided they are carefully chosen [4].

Therefore, some research exists that considers reducing the number of columns in Tab. 1. All these studies take for granted that a common set of topics has to be used. Indeed, this seems quite reasonable: if two different systems (or runs) are evaluated on two different topics, the effectiveness of each system will depend on topics features, and the evaluation will be unfair to the system being evaluated on a difficult topic (see [5] for a detailed discussion of this issue).

Although related approaches do exist [10,4], we are not aware of any research explicitly trying to evaluate using *different topics for different runs*. This short paper is a first attempt to study how to reduce the number of cells in Tab. 1 without the constraint of staying on the same columns. Following [4], we seek for the best-case bound, and we concentrate on the potential limit, not on an actual way to reach it (although we hint at some sensible strategies).

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 342–345, 2009.

Table 1. AP, MAP and AAP

	t_1	$\cdots$	t_n	MAP
r_1	$\mathrm{AP}(r_1, t_1)$	$\cdots$	$\mathrm{AP}(r_1, t_n)$	$\mathrm{MAP}(r_1)$
$\vdots$		$\ddots$		$\vdots$
r_m	$\mathrm{AP}(r_m, t_1)$	$\cdots$	$\mathrm{AP}(r_m, t_n)$	$\mathrm{MAP}(r_m)$
AAP	$\mathrm{AAP}(t_1)$	$\cdots$	$\mathrm{AAP}(t_n)$	

Of course, by using different topics for different runs, the total number of used topics might turn out to be higher than when using the same topics for all runs. Moreover, it is intuitive that judging a new document on the same — and known — topic costs less effort to a human assessor than judging a new document on a novel topic (and this fact is exploited by assessors in TREC-like initiatives). Therefore the number of topics (i.e., columns in Tab. 1) used seems a more accurate measure of assessment effort than the number of topic-document pairs (i.e., cells). However, it is interesting to study this alternative, especially because novel crowdsourcing-based assessing techniques are being proposed [2,1]: with these novel approaches, a crowd of assessors can judge single topic-document pairs, with no notion of "staying on the same topic to reduce the effort".

2 Methods and Data

We use TREC 8 data (129 runs and 50 topics); as done in [9,4], we remove the worst 25% runs, thus keeping 96 of them. We assume the distribution of MAPs of runs, obtained by taking the arithmetic means of each row in Tab. 1, over all the fifty columns, as ground truth, and we study how to approximate it by using fewer AP values, or their normalizations, on each row. As stated above, we do not require that the same set of topics is used for all the runs.

2.1 Normalizations

Some normalizations based on topic difficulty have recently been proposed, and are used in the following to take into account topic difficulty and to avoid the above described unfairness.

In [6] the average of AP values (over one column in Tab. 1) is named *Average AP* (AAP). It is used to measure topic ease (on the basis of the TREC runs) and to define the $\overline{\mathrm{AP}_\mathrm{A}}(r_i, t_j) = \mathrm{AP}(r_i, t_j) - \mathrm{AAP}(t_j)$ normalization. On the basis of this normalization, the metric $\overline{\mathrm{MAP}}(r_i) = \frac{1}{n}\sum_{j=1}^{n} \overline{\mathrm{AP}_\mathrm{A}}\,(r_i, t_j)$ is defined; it is shown that $\overline{\mathrm{MAP}}\,(r_i)$ is equivalent to $\mathrm{MAP}\,(r_i)$. Webber and colleagues propose the standard $\frac{x-\mu}{\sigma}$ normalization (where μ is the mean, here AAP, and σ is the standard deviation) [10]. In [5] a different metric, named NMAP, is proposed. It, on purpose, measures something slightly different, penalizing bad performance on easy topics and awarding good performance of difficult topics.

We also consider MAP without any normalization, thus we have $\overline{\mathrm{MAP}}$, Webber, NMAP, and MAP.

2.2 Strategies

We define four *strategies* to build $T_c(r_i)$, i.e., the topic set of cardinality c (for increasing $c = 1..50$) for each run r_i:

S1. *Random.* For each c, $T_c(r_i)$ is built by taking c random AP values among the not yet chosen $51 - c$. To avoid dependencies on the single values chosen, we repeat the process and average the results.

S2. *Pseudo-optimal.* For each c, $T_c(r_i)$ is built starting from the previous topic set $T_{c-1}(r_i)$ and adding to it the topic (among the remaining $51 - c$) such that the obtained $\mathrm{MAP}(T_c(r_i))$ is the closest to MAP. $T_0 = \emptyset$.

S3. *Difficult-to-Best (DtB).* A sensible criterion is to assign more difficult topics to more effective runs and easier topics to less effective runs [5]. According to this criterion, $T_c(r_i)$ is built by: (i) sorting the runs by MAP, from the least effective to the most effective; (ii) sorting the topics by AAP, from the least difficult to the most difficult; and (iii) assigning, in order, to the most effective systems the most difficult (and not yet assigned) topic, and so on.

S4. *Easy-to-Best (EtB).* The symmetric strategy of S3.

3 Results and Discussion

We present correlation curves similar to those in [4]. For each c value (x axis) we represent the correlation between the distributions obtained by various measures defined over T_c and the MAP distribution. Fig. 1 (left) shows seven Kendall's correlation curves. Three curves (best, average, and worst) are obtained in [4], and are correlation values obtained under the *same-topics* constraint (selecting topics on the same columns). The other curves are *different-topics*, i.e., they are obtained without the same-topics constraint, using S1. S1 is very similar to average when combined with $\overline{\mathrm{MAP}}$ and Webber normalizations, whereas NMAP and MAP have lower than average correlations.

Fig. 1 (right) shows that with a more careful strategy choice like S3, correlations for Webber and $\overline{\mathrm{MAP}}$ increase, and also NMAP is much higher than average. MAP (which is not normalized) increases, but also becomes much more unstable. This proves that when working on different topics for different systems

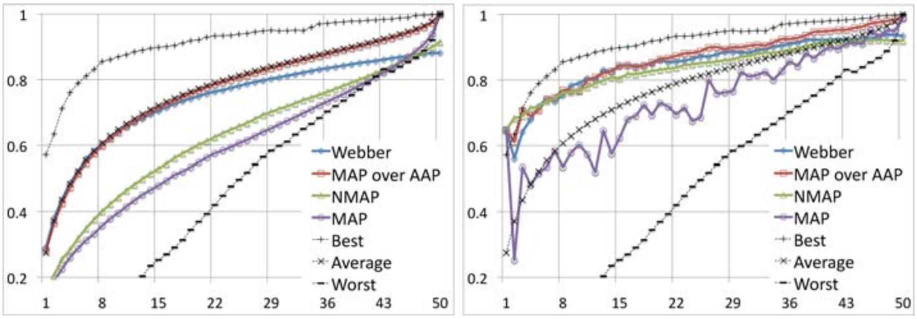


Fig. 1. Kendall's correlations for S1 (left) and S3, or DtB, strategies (right)

it is possible to define strategies to get correlations higher than average, and not far from the optimal that can be obtained with common topics (the higher best curve in figure). This is confirmed by S4 correlation values that, when computed on $\overline{\text{MAP}}$, give a curve lower than average, and rather close to worst. Pearson's correlation, not shown for space limitations, gives the same results.

Finally, one might wonder what is the maximum correlation, under *different-topics*: it is quite high. Although S2 is not optimal, it reaches, for any c value and for MAP (thus without any normalization), a Kendall's correlation of at least 0.91 and a Pearson's correlation of at least 0.993: its correlation curve stays above the curves in the figures.

To summarize the results obtained, our analysis shows that it is theoretically possible to evaluate IR effectiveness, within the TREC paradigm, on different topics for different systems. Also, when appropriate normalizations and strategies are used, it seems possible to evaluate effectiveness on much less data than the usual 50 topics used in TREC. However, this last statement needs further confirmation, since when using the above defined strategies, several topics (columns) end up used even for low c values. Finally, outside TREC, in interactive studies, the different-topics situation is the rule: each subject interprets in different ways the induced needs (when needs are framed by the experimenter) and/or each user comes with his/her own need (when real needs are used).

Acknowledgements. We thank the anonymous reviewers for useful comments.

References

1. Alonso, O., Mizzaro, S.: Relevance criteria for e-commerce: A crowdsourcing-based experimental analysis. In: 32nd SIGIR (2009) (in press)
2. Alonso, O., Rose, D., Stewart, B.: Crowdsourcing for relevance evaluation. SIGIR Forum 42(2), 9–15 (2008)
3. Buckley, C., Voorhees, E.: Evaluating evaluation measure stability. In: 23rd SIGIR, pp. 33–40 (2000)
4. Guiver, J., Mizzaro, S., Robertson, S.: A few good topics: Experiments in topic set reduction for retrieval evaluation. In: ACM TOIS (2009) (in press)
5. Mizzaro, S.: The Good, the Bad, the Difficult, and the Easy: Something Wrong with Information Retrieval Evaluation? In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 642–646. Springer, Heidelberg (2008)
6. Mizzaro, S., Robertson, S.: HITS hits TREC: exploring IR evaluation results with network analysis. In: 30th SIGIR, pp. 479–486 (2007)
7. Sanderson, M., Zobel, J.: Information retrieval system evaluation: effort, sensitivity, and reliability. In: 28th SIGIR, pp. 162–169 (2005)
8. Sparck Jones, K., van Rijsbergen, C.J.: Information retrieval test collections. Journal of Documentation 32, 59–75 (1976)
9. Voorhees, E., Buckley, C.: The effect of topic set size on retrieval experiment error. In: 25th SIGIR, pp. 316–323 (2002)
10. Webber, W., Moffat, A., Zobel, J.: Score standardization for inter-collection comparison of retrieval systems. In: 31st SIGIR, pp. 51–58 (2008)
11. Zobel, J.: How reliable are the results of large-scale information retrieval experiments? In: 21st SIGIR, pp. 307–314 (1998)

An Ad Hoc Information Retrieval Perspective on PLSI through Language Model Identification[*]

Jean-Cédric Chappelier and Emmanuel Eckard

School of Computer and Communication Sciences
École Polytechnique Fédérale de Lausanne, CH–1015 Lausanne

Abstract. This paper proposes a new document–query similarity for PLSI that allows queries to be used in PLSI without folding-in. We compare this similarity to Fisher kernels, the state-of-the-art approach for PLSI, on a corpus of 1M+ word occurrences coming from TREC–AP.

1 Introduction

Ten years ago, the Probabilistic Latent Semantic Indexing (PLSI) model was proposed [4], considering documents as mixture proportions of latent-topics: the probability of occurrence of a pair (d, w) of a document model d and a term w is modelled as $P(d, w) = \sum_{z \in Z} P(z)\, P(w|z)\, P(d|z)$.

To cope with the non-generative nature of PLSI regarding to new documents, a scheme named *folding-in* was proposed [4,3]. This technique estimates the parameters $P(d|z)$ for unknown documents d, such as queries in an ad hoc Information Retrieval (IR) framework. These are learnt by a simplified process that engenders problems such as adequacy with the $P(d|z)$ coming from the training set, or inaccuracies in log-likelihood estimation of the test set [8].

This paper introduces a document–query similarity based on language model identification [7,9] that entirely avoids folding-in. Its performances are compared to Fisher kernels, the state-of-the-art similarities for PLSI [5,2].

2 Avoiding Folding-in of Queries

To entirely dispense with folding-in in PLSI, we propose a method inspired from language modelling: queries are no longer considered as new document models for which new parameters $P(d|z)$ must be learnt, but rather as new *occurrences* of already learnt document models. Rather than inferring a new model for the queries, the retrieval problem turns into model identification: for a given query q, which are the known models d best representative of q?

Traditional answers include maximisation of the query log-likelihood $\mathcal{S}_{\text{LogL}}$ [7] and minimisation of the Kullback-Leibler (KL) divergence $\mathcal{S}_{\text{KL}}$ between the empirical distribution of q and the model distribution of d [6]:

[*] The hereby described work has been supported by the Swiss National Science Foundation doctoral grants #200021-111817 and #200020-119745.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 346–349, 2009.

$$\mathcal{S}_{\mathrm{LogL}}(d,q) = \sum_{w \in q \cap C} n(q,w) \log P(d,w) \ , \ \mathcal{S}_{\mathrm{KL}}(d,q) = \sum_{w \in q \cap C} \widehat{P}(w|q) \log \frac{P(w|d)}{\widehat{P}(w|q)} \ ,$$

where $n(q,w)$ is the number of occurrences of word w in query q; $\widehat{P}(w|q) = n(q,w)/|q|$, its normalisation by the length $|q|$ of query q; and where "$w \in q \cap C$" denotes all the words appearing in q such that $P(d,w) > 0$.[1]

When performing retrieval, i.e maximising $\mathcal{S}(d,q)$ w.r.t. d for a given query q, the two approaches differ by an additive factor of "$|q| \ \log P(d)$" since

$$\mathcal{S}_{\mathrm{KL}}(d,q) = \frac{1}{|q|}\left(\mathcal{S}_{\mathrm{LogL}}(d,q) - |q| \ \log P(d)\right) - \sum_{w} \widehat{P}(w|q) \log \widehat{P}(w|q) \ .$$

Both $\mathcal{S}_{\mathrm{KL}}$ and $\mathcal{S}_{\mathrm{LogL}}$ can use any estimator of $P(w|q)$ smoother than $\widehat{P}(w|q)$ [11]. For instance, we can consider Jelinek-Mercer smoothing to illustrate the point:

$$\widetilde{P}(w|q) = (1-\lambda)\,\widehat{P}(w|q) + \lambda\, P_{\mathrm{GE}}(w) \ , \qquad P_{\mathrm{GE}}(w) = \frac{\sum_{d \in C} n(d,w)}{\sum_{d \in C} \sum_{w \in d} n(d,w)} \ ,$$

with $\lambda \in [0,1]$, and $P_{\mathrm{GE}}(w)$ the collection language model.

Another way to build a smoothed estimator is pseudo-feedback [1,10]: a first retrieval is performed and the N best retrieved documents (for small N) are used to estimate $\widetilde{P}(w|q) = \frac{1}{N}\sum_{i=1}^{N} P\left(d_i(q), w\right)$, with $d_i(q)$ the i^{th} best match. A second and final retrieval phase is then performed. The first retrieval can use either raw word document frequencies, or smoothed estimators.

This leaves us with 8 possible document similarity measures: query log-likelihood $\mathcal{S}_{\mathrm{LogL}}$ or KL divergence $\mathcal{S}_{\mathrm{KL}}$, and for both, possibilities to use Jelinek-Mercer smoothing, pseudo-feedback, or both. We compared these 8 models with the three best Fisher kernel variants, K^{H}, K_w^{H}, and $K_w^{\mathrm{DFIM\text{-}H}}$ [2].

3 Experiments

The non-generative nature of PLSI for unknown document models requires parameter estimation on the *entire* document collection to be evaluated. PLSI can thus hardly be used on TREC-sized collections. In line with previously published work on PLSI, we used the standard IR benchmarks from the SMART collections (CACM, CISI, MED, CRAN and TIME). We furthermore explored the limits of PLSI learning tractability using a significantly bigger corpus (over 5 times as many documents and 10 times as many word occurrences as in the biggest SMART collections) consisting of a subpart of the TREC–AP 89 corpus.We kept the 7466 first documents of this collection[2], and queries 1 to 50. For the experiments on the SMART collections, we performed 6 runs with different

[1] Notice that $P(d,w) > 0$ does not necessarily imply $w \in d$.

[2] Documents AP890101-0001 to AP890131-0311. The EM learning for $|Z| = 128$ took 45 hours of CPU time and used 6.7 GB of RAM on a dedicated computation server with one octo core 2-GHz Intel Xenon processor and 32 GB of memory.

Table 1. Main results and conclusions of experiments over 11 models on 6 corpora

		CACM	CRAN	TIME	CISI	MED	AP89_01XX
Results	BM25 MAP	**31.4**	**42.4**	**69.2**	12.3	52.3	19.7
	Best PLSI model MAP	30.0	39.6	60.8	**20.2**	**53.8**	**21.6**
	Best PLSI model is:	K_w^{H}	$\mathcal{S}_{\mathrm{KL}}$	$K_w^{\mathrm{DFIM\text{-}H}}$	K_w^{H}	K^{H}	$K_w^{\mathrm{DFIM\text{-}H}}$
	for $\lvert Z\rvert =$	16	128	8	8	32	48
	K_w^{H} MAP	**30.0**	33.6	55.6	**20.2**	49.8	16.5
	$K_w^{\mathrm{DFIM\text{-}H}}$ MAP	23.2	37.0	**60.8**	15.6	45.5	**21.6**
	$\mathcal{S}_{\mathrm{KL}-128}$ MAP	22.9	**39.6**	49.1	19.5	**52.8**	11.4
Concl.	$\mathcal{S}_{\mathrm{KL}-128}$ w.r.t. Fisher kernels	$<$	$>$	$<$	$\simeq$	$\simeq$	$<$
	PLSI > BM25?	No	No	No	**YES**	**yes**	**yes**
	Does smoothing help?	No	No	No	No	No	No

learning initial conditions for all models, and for different numbers of topics: $|Z| \in \{1, 2, 8, 16, 32, 64, 128\}$. For the TREC-AP part, we performed a single run for each $|Z| \in \{1, 32, 48, 64, 80, 128\}$. For all the experiments, we performed stemming using the Porter algorithm of Xapian. Results were obtained using the standard `trec_eval` tool.

The main results out of these experiments, summarised in Table 1 and Fig. 1, are: (1) $\mathcal{S}_{\mathrm{KL}}$ performs much better than $\mathcal{S}_{\mathrm{LogL}}$, both have a growing performance with $|Z|$, the number of latent-topics; (2) $\mathcal{S}_{\mathrm{KL}}$ can outperform the best Fisher kernel on CRAN, and reaches similar performances on MED and CISI; (3) neither Jelinek-Mercer smoothing nor pseudo-feedback improve performance. Furthermore, smoothing significantly increases the runtime of evaluation: rather

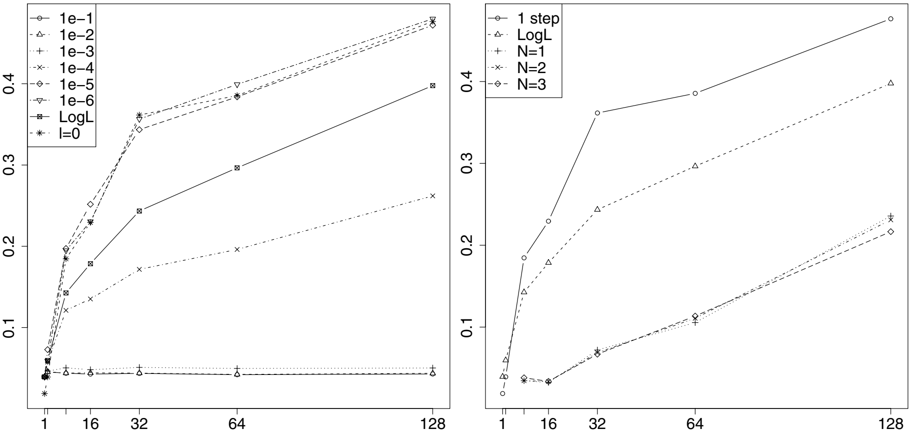


Fig. 1. A typical example (MAP vs $|Z|$) on TIME illustrating how $\mathcal{S}_{\mathrm{KL}}$ outperforms $\mathcal{S}_{\mathrm{LogL}}$, and how smoothed (left) or pseudo-feedback retrieval (right) estimates for $P(q|w)$ degrade the $\mathcal{S}_{\mathrm{KL}}$ performances compared to raw estimate $\widehat{P}(q|w)$ ("l=0" on the left and, "1 step" on the right), for different values of λ (left) and N (right)

than involving only those terms appearing in both the query and the document, all the terms of the vocabulary have to be considered. Evaluation is between 2 (CISI) and 20 (TIME, MED) times slower for smoothing, and between 30 (MED) and 150 (CRAN) times slower for pseudo-feedback with $N = 3$.

4 Conclusion

We introduce a new document similarity for PLSI, based on language model identification, which entirely avoids query folding-in. It is evaluated in an IR framework on a collection larger than the SMART collections on which PLSI is usually evaluated. The main conclusions are that (1) language model identification can compete with the best Fisher kernel variants, especially for high number of topics; (2) either KL divergence or Fisher kernels can compete with BM25, especially on semantically tougher corpora like CISI, MED or TREC–AP; (3) however, neither log-likelihood similarity nor any simple smoothing method of $\widehat{P}(w|q)$ improved the results. Out of the 8 models here studied, only the KL divergence between $P(w|d)$ and raw $\widehat{P}(w|q)$ turned out to be interesting.

References

1. Cadez, I.V., Gaffney, S., Smyth, P.: A general probabilistic framework for clustering individuals and objects. In: Proc. of 6th KDD, pp. 140–149 (2000)
2. Chappelier, J.-C., Eckard, E.: PLSI: the true Fisher kernel and beyond. In: Proc. of ECML/PKDD (2009)
3. Hinneburg, A., Gabriel, H.-H., Gohr, A.: Bayesian folding-in with Dirichlet kernels for PLSI. In: Proc. of 7th Int. Conf. on Data Mining, pp. 499–504 (2007)
4. Hofmann, T.: Probabilistic latent semantic indexing. In: Proc. of 22th Int. Conf. on Research and Development in Information Retrieval (SIGIR), pp. 50–57 (1999)
5. Hofmann, T.: Learning the similarity of documents. In: Adv. in Neural Information Processing Systems, vol. 12, pp. 914–920 (2000)
6. Lafferty, J., Zhai, C.: Document language models, query models, and risk minimization for information retrieval. In: Proc. of 24th Annual Int. Conference on Research and Development in Information Retrieval (SIGIR), pp. 111–119 (2001)
7. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proc. of 21st SIGIR, pp. 275–281 (1998)
8. Welling, M., Chemudugunta, C., Sutter, N.: Deterministic latent variable models and their pitfalls. In: SIAM Conference on Data Mining SDM 2008 (2008)
9. Zhai, C.: Statistical language models for information retrieval: A critical review. Foundations and Trends in Information Retrieval 2(3), 137–213 (2008)
10. Zhai, C., Lafferty, J.: Model-based feedback in the language modeling approach to information retrieval. In: Proc. of 10th CIKM, pp. 403–410 (2001)
11. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst. 22(2), 179–214 (2004)

Less is More: Maximal Marginal Relevance as a Summarisation Feature

Jan Frederik Forst, Anastasios Tombros, and Thomas Roelleke

Department of Computer Science
Queen Mary, University of London
{frederik,tassos,thor}@dcs.qmul.ac.uk

Abstract. Summarisation approaches aim to provide the most salient concepts of a text in a condensed representation. Repetition of extracted material in the generated summary should be avoided. Carbonell and Goldstein proposed Maximal Marginal Relevance as a measure to increase the diversity of documents retrieved by an IR system, and developed a summariser based on MMR. In this paper, we look at the viability of MMR as a feature in the traditional feature-based summarisation approach proposed by Edmundson.

1 Introduction and Background

Summarisation approaches have been successfully applied to a range of IR tasks: to reduce the indexed representation of documents, to indicate the usefulness of retrieved documents, or to inform users on specific topics.

In particular, *feature-based* summarisation has shown a consistently good performance at summarising documents, and has proved to be a main-stay in summarisation research (see [4] for an overview). However, similar sentences in the source document(s) will receive similar salience scores, leading to repetition of extracted material in the generated summary. A similar problem exists in the context of document retrieval, where a query with multiple interpretations needs to be interpreted by the retrieval systems such that the returned documents fulfil a user's information need. If the system's interpretation overlaps with the user's intended information need, she will see many relevant documents; otherwise, the system will return many non-relevant documents, and only very few (or none) relevant ones.

To overcome this problem, Carbonell and Goldstein proposed Maximal Marginal Relevance (MMR) – a measure where the retrieval status value (RSV) of a document is influenced by other already retrieved documents: documents similar to retrieved documents have their RSV lowered, thus boosting dissimilar documents [1]. In addition, they applied MMR to summarisation, developing a system where traditional features where incorporated into the MMR framework. However, this approach makes it difficult to distinguish between the influence of summarisation features, and the overall MMR framework. We therefore discuss how an MMR-like feature can be incorporated into a feature-based summariser, and evaluate it on two test collections.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 350–353, 2009.

2 Maximal Marginal Relevance

Carbonell and Goldstein motivate MMR with the need to include novelty into the ranking of documents to prevent the presentation of partially or fully duplicate information. Combined with the traditional notion of relevance, they propose the new metric "relevant novelty", which can be measured as MMR, using a weighted linear combination of relevance and novelty [1]:

$$\text{MMR} := \arg \max_{D_i \in R \setminus S} [\lambda(\text{Sim}_1(D_i, Q)) - (1 - \lambda)(\max_{D_j \in S} \text{Sim}_2(D_i, D_j))] \tag{1}$$

where R is a ranked list of documents, S is the set of documents in R already retrieved, and Sim_1 and Sim_2 are similarity measures, which can be the same, or can be set to different similarity metrics. Adjusting the value of λ allows a readjustment of the behaviour of MMR: with a setting $\lambda = 1$, MMR behaves like a traditional ad-hoc model, while it reduces to a maximal diversity ranking with a setting $\lambda = 0$.

2.1 MMR-Based Summarisation

Goldstein *et al.* proposed a multi-document summarisation system based on MMR, MMR-MD [2]. MMR-MD implements the MMR approach by modelling Sim_1 as a weighted combination of query overlap, coverage, content, and time-sequence, whereas Sim_2 is modelled by a weighted combination of different sentence overlaps.

However, it does not become clear how much the diversity measure of MMR beneficially influences the extraction of sentences. We therefore included MMR into an existing summariser as an individual feature, where only the diversity component is considered for weighting sentences (i.e. setting λ to 0). This maximum diversity rank is then integrated with other features to give an overall sentence score. Using this setup allows us to more carefully evaluate the benefit of MMR for the purpose of summarisation.

3 Experiments

To estimate the usefulness of MMR as a summarisation feature, we first developed our version of the MMR diversity rank, and then included it into a summarisation model with other, more traditional features. We then used this MMR-augmented combination of features to generate summaries for two test collections, and compared the results to those obtained by using the traditional features only. Additionally, we created summaries using each of the features in isolation.

3.1 MMR Feature

A diversity ranking approach should rank sentences according to their dissimilarity to a set of other sentences: the more dissimilar they are, the higher they should rank. To model this diversity ranking scheme, we compare a sentence to each of the sentences in a given set. The diversity rank of a sentence with respect to this set is the sum of TF-weights of all terms which do *not* occur in the sentence it is compared to.

To compensate differences in the length of sentences, and the size of sets to which the sentences are compared, we normalise the per-sentence comparisons by the sentence length, and the overall set-comparison by the size of the set. The MMR-weight of a sentence s with respect to a set of sentences S is thus defined as:

$$MW(s,S) := \frac{\sum_{m \in S} \frac{\sum_{t \in s, t \notin m} (P_L(t,s))}{|s|}}{|S|} \tag{2}$$

where $P_L(t,s)$ is the TF-weight of a term t in sentence s. There are (at least) two possible approaches for incorporating such an MMR feature into a summariser. The MMR feature can either be used as an additional feature amongst other features (such as location, term-frequency etc.), or can be used as a *reweighting* stage after sentences were weighted using traditional features. In the former approach, the MMR weight of a sentence is determined w.r.t. all other candidate sentences, while in the latter approach candidate sentences weighted by the traditional features only are *reweighted* by their similarity w.r.t. the set of already selected summary sentences.

3.2 Experimental Setting

In addition to the MMR feature, our summariser also used the Key-, Title-, Location-, and Query-feature (referred to here as KTLQ; see [4] for details).

To evaluate the different summarisation models, we applied them to two test collections: the Document Understanding Conference (DUC) AQUAINT corpus, and the INEX 2004 document collection. Automatic summaries were generated for both collections by all summarisers, and were compared to human-authored reference summaries. While the DUC corpus provides reference summaries for evaluation, no such reference summaries were available for the INEX collection. Instead, we used the original abstracts of documents in the collection as a reference. Furthermore, we used INEX document titles as summarisation queries; to avoid a bias in the generated summaries, we did not use the Query-feature for the INEX runs. To evaluate the quality of the summaries we generated with the provided reference summaries, we used the ROUGE evaluation framework, and calculated the ROUGE-1 Precision and Recall scores [3].

4 Results

The averaged results in Table 1 show that the Query-feature outperforms all other features for the DUC collection, both on precision and recall. A comparison of the performance of the different summarisers shows a less clear picture, with MMR as a reranking feature exhibiting only marginal improvements.

For INEX, Table 2 shows that the Title-feature performs better than the MMR-, Location-, or Key-feature. A comparison of the different summarisation systems shows that all three combinations of features display a very similar performance. While the feature combination using MMR as a reweighting approach can exhibit a slight advantage for recall, all systems become virtually indistinguishable for precision.

Table 1. DUC: ROUGE1 recall- and precision scores for features and systems

Feature	Recall	Precision
Key	0.34219	0.33140
Title	0.35979	0.34770
Location	0.33829	0.32061
Query	**0.38180**	**0.36242**
MMR	0.33426	0.31870

System	Recall	Precision
KTLQ	0.38468	0.36455
MMR-as-Feat.	0.38425	0.36307
MMR-rerank	**0.38488**	**0.36562**

Table 2. INEX: ROUGE1 recall- and precision scores for features and systems

Feature	Recall	Precision
Key	0.65171	0.11260
Title	**0.68971**	**0.12007**
Location	0.65165	0.11258
MMR	0.65420	0.11220

System	Recall	Precision
KTL	0.69229	0.12060
MMR-as-Feat.	0.69453	0.12085
MMR-rerank	**0.69640**	**0.12128**

5 Conclusion

In this paper, we looked at the use of Maximal Marginal Relevance (MMR) as a summarisation feature. We compared the MMR feature to other, traditional, summarisation features, and evaluated its performance on two test collections: the DUC collection for multi-document summarisation, and the INEX collection for single document summarisation.

The evaluation results show that although the MMR feature itself does not perform particularly well, a summarisation system implementing an MMR feature performs better than comparable systems without the MMR feature. However, the performance difference between the MMR- and the non-MMR-versions of the summarisation systems were very small, making a real-world benefit of the computationally intensive MMR approach questionable.

References

1. Carbonell, J., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: SIGIR 1998, pp. 335–336. ACM, New York (1998)
2. Goldstein, J., Mittal, V., Carbonell, J., Callan, J.: Creating and evaluating multi-document sentence extract summaries. In: CIKM 2000, pp. 165–172. ACM, New York (2000)
3. Lin, C.-Y.: ROUGE: A Package for Automatic Evaluation of Summaries. In: Proceedings of the ACL 2004 Workshop, Barcelona, Spain, July 2004, pp. 74–81. ACL (2004)
4. Tombros, A., Sanderson, M.: Advantages of query biased summaries in information retrieval. In: SIGIR 1998, pp. 2–10. ACM, New York (1998)

On the Notion of "An Information Need"

Eduard Hoenkamp

University of Maastricht
hoenkamp@acm.org

Abstract. 'Information need' is a notion in IR that is ubiquitous, important, and intuitively clear. So far, surprisingly, the term seems to have defied formal definition. Of course, IR can continue to prosper without a formalization of 'information need'. Yet when a field gets more mature there comes a time that frequently used notions should be formalized to make them susceptible of scrutiny. For IR such formalization should (1) be independent of a particular query language or document model, (2) allow that users formulate a need for information that may be unavailable or even nonexistent, and (3) allow that users try to circumscribe the very information they do not possess. To this end, the paper uses lattice theory to define a 'formal information need', which, we argue, coincides with the intuitive notion precisely when a user's need for information can actually be filled.

1 Introduction

When we see ourselves or others look up information on the world wide web, we can make three observations: First, people can use the same query but may be looking for different information. Second, they may be looking for the same information but use different queries. And third, they usually refine their queries to zoom in on their *information need*. A definition of that term, however, is quite elusive. It is something we hold in our heads and of which it is not clear that it can ever be fully satisfied. What is clear, though, is that information needs can often be discerned by the different queries that are issued, as well as by the different documents people subsequently select. That is, queries and document sets seem to act as dual descriptions of the same information need. It is difficult, however, to maintain that the two descriptions are completely isomorphic. Hence we will use a relation that is sufficient yet weaker than isomorphism, called a *Galois Connection*, for which we consider queries and relevant documents as a pair instead of choosing one or the other in isolation.

2 Mapping Queries to Documents and Back

The formalization that follows was inspired by formal concept analysis (FCA), and so perhaps an example from that subject is helpful.

FCA is an approach to meaning representation that describes concepts as a binary relation between 'objects' and 'attributes' [1]. For example, a horse can

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 354–357, 2009.

be described as an object that represents the concept of "horse" as a set of attributes, such as solid-hoofed, herbivorous, mammal, or perhaps attributes such as equine, domesticated, etc. A smaller set of attributes could define the more general concept of equines, subsuming zebra and donkey. A larger set of attributes defines a more specific concept, e.g. stallion or foal. This way, specificity as an order relation defines a lattice over the attributes. The objects themselves can be ordered by set inclusion, and as more attributes are associated with fewer objects the order is reversed. This kind of association between partially ordered sets (posets), turns up in many branches of mathematics, and many of its properties were studied by Ore [2] who named the association a *Galois connection*:

Definition 1. *Consider posets* $(P, \preccurlyeq_p)$ *and* $(Q, \preccurlyeq_q)$ *and functions* $f : P \to Q$ *and* $g : Q \to P$*. Then* $\langle f, g \rangle$ *is called a* **Galois connection***, if for all* $x \in P$ *and* $y \in Q$*:*

$$f(x) \preccurlyeq_q y \Longleftrightarrow g(y) \preccurlyeq_p x$$

Readers unfamiliar with the subject can find a brief overview of Galois connections, with many examples in [3]. Because of the similarity with FCA, we will adopt some of its terminology. In FCA, P in definition 1 would be all sets of attributes $Attr$ (i.e. 2^{Attr}) and Q all sets of objects Obj each ordered by set inclusion. Then the attributes correspond to the objects and vice versa via mappings from one to the other:

$$2^{Attr} \underset{\mathbf{g}}{\overset{\mathbf{f}}{\rightleftharpoons}} 2^{Obj}$$

A *formal concept* is a pair $\langle A, B \rangle$, with $A \subseteq Attr$ and $B \subseteq Obj$, for which $f(A) = B$ and $g(B) = A$. A is called the *intent*, and B the *extent* of the concept. We will now explore how the FCA approach translates to Information Retrieval.

2.1 The Galois Connection in Information Retrieval

Let us apply definition 1 such that it can be used in information retrieval. The set P will be the query language over the features that we find appropriate for the domain. An example could be words (terms) for text retrieval, or texture, color, and shape in the case of image retrieval. The expressions in the query language can be ordered e.g. by set inclusion. But they could also form a Lukaciewicz logic such as Boolean or Fuzzy logic, ordered as usual by meet and join of the expressions. Similarly, the Q in definition 1 could be the subsets of the corpus, usually sets of documents ordered by set inclusion. An instance would be Salton's early definition of a retrieval system (on page 211 of [4]), where P is called the 'request language', and f the 'retrieval function'. (But note that this definition had disappeared in the second edition of Salton's book.) For this paper, we use **match** and **index**, so coined in [5], for f and g in defintion 1. So with these definitions, the pair $\langle \textbf{match}, \textbf{index} \rangle$ will be the Galois connection that represents the relationship between query language and corpus. In FCA the term 'concept' as something we have in our minds, is superseded by a well-defined *formal concept*. Analogously we will define a *formal information need* to supersede the vaguer notion of 'information need':

Definition 2. *Given (1) a lattice over query language $\mathcal{L}(T)$ of terms T, (2) a corpus D of documents, and (3) match function **match** and index function **index**, for which*

$$\mathcal{L}(T) \underset{\textbf{\textit{index}}}{\overset{\textbf{\textit{match}}}{\rightleftharpoons}} 2^D$$

A **formal information need** *is a pair $\langle A, B\rangle$, with $A \subseteq \mathcal{L}(T)$ and $B \subseteq D$, for which $B = \textbf{\textit{match}}(A)$ and $A = \textbf{\textit{index}}(B)$.*

So the query is the *intent* of the formal information need, and the *extent* is a set of documents. A straightforward example of **match** and **index** is what the names suggest: **match** maps queries to the document sets, and **index** indexes the documents. Note that one is not an inverse of the other, but they determine each other uniquely as follows [3]:

Theorem 1. ***match*** *and* ***index*** *are quasi-inverses, i.e.*
$\textbf{\textit{match}} \circ \textbf{\textit{index}} \circ \textbf{\textit{match}} = \textbf{\textit{match}}$ *and* $\textbf{\textit{index}} \circ \textbf{\textit{match}} \circ \textbf{\textit{index}} = \textbf{\textit{index}}$

This expresses the intuition about fulfilling an information need: Suppose (1) you take all the documents matching a query (**match**), and (2) you collect all the queries that could have produced these documents (**index**). If (3) no new queries are added to the originals, then the queries and documents are just two different representations of the same information need. The theorem entails that first, **match**∘**index** is idempotent, and second, **match**∘**index**∘**match** produces all and only the documents that fulfill the information need.

Some concrete examples of the definition. Take Google's "I'm feeling lucky" button: If the user types a query q_1, only the top ranked document d_1 is returned, and the formal information need would be $\langle\{q_1\}, \{d_1\}\rangle$. A few years back we looked up 'search' and 'Google' only to find $\langle\{\textit{search}\}, \{\rightarrow \textit{AltaVista's home}\}\rangle$ and $\langle\{\textit{google}\}, \emptyset\rangle$. Notice that even as these are apparently formal information needs in the Galois connection of Google's $\langle\textbf{match}_{\textbf{lucky}}, \textbf{index}\rangle$, they were not our own information need. Indeed, the definition should allow for it, as it should be independent of a particular search technique or query language.

3 Recovering the Connection through Feedback

There are several approaches to tighten the separation between formal and informal information need: (1) *change the Galois connection* by altering the **match** function. An obvious example is using WordNet[6] to take synonymity of terms as an equivalence relation over the queries. Relevance feedback with reweighting (e.g. [7]) is another example. (2) *Leave the Galois connection intact* as in the everyday use of a browser: this only changes the query (i.e. the intent) until a formal information need is found that has the needed information as extent. Note that even if more documents could fulfill the information need, the idempotence of **match** ∘ **index** expressed in theorem 1 guarantees that the information need coincides with a formal information need. We have applied the theory described above to a system that provides visual feedback of search results to zoom in on an information need [8]. The system constructs a Galois connection between the document space and and a visualization of documents as points on a sphere. Now the following property of Galois connections can be used [3]:

Theorem 2. *If* $P \underset{g}{\overset{f}{\rightleftharpoons}} Q$ *and* $Q \underset{i}{\overset{h}{\rightleftharpoons}} R$ *are Galois connections, then so is* $P \underset{g \circ i}{\overset{h \circ f}{\rightleftharpoons}} R$*.*

Theorem 2 therefore tells us that there is also a Galois connection between the input language and the points on the sphere. That is, one can start with a query to the define the extent, or on the sphere to define the intent of the Galois connection. Either way, it will result in a formal information need that approaches the informal information need.

4 Conclusion

We proposed a formal definition of the ubiquitous notion of 'information need', to capture the way it is normally understood. Alternative approaches might focus on the notion of similarity, perhaps formalized in terms of a topology (starting with a T_0 space) or category theory (using upper and lower adjuncts). We maintained that approaching the relationship between query language and document set as an isomorphism is definitely too strong, and the weaker notion of a Galois connection seems just right. We showed how it can be used to show that some manipulations, such as visual feedback, are well-founded, and we think that it may help to define some issues in IR in a more precise and perspicuous way.

References

1. Ganter, B., Wille, R.: Formal concept analysis: mathematical foundations. Springer, Berlin (1999)
2. Ore, O.: Galois connexions. Trans. American Math. Society 55, 493–513 (1944)
3. Erné, M., Koslowski, J., Melton, A., Strecker, G.E.: A primer on Galois connections. In: Todd, A.R. (ed.) Papers on general topology and applications. Annals of the New York Academy of Sciences, vol. 704, pp. 103–125 (1993)
4. Salton, G.: Automatic Information organization and Retrieval. McGraw-Hill, New York (1968)
5. Grootjen, F.A., van der Weide, T.P.: Dualistic ontologies. International Journal of Intelligent Information Technologies 1(3), 34–55 (2005)
6. Miller, G.: Wordnet: A lexical database for english. Comm. of the ACM 38(11), 39–41 (1995)
7. Robertson, S.E., Walker, S., Hancock-Beaulieu, M., Gull, A., Lau, M.: Okapi at TREC. In: Text REtrieval Conference, pp. 21–30 (1992)
8. Hoenkamp, E., van Dinther, G.: Live visual relevance feedback for query formulation. In: Proceedings of SIGIR 2005, pp. 611–612 (2005)

A Logical Inference Approach to Query Expansion with Social Tags

Christina Lioma[1], Roi Blanco[2], and Marie-Francine Moens[1]

[1] Computer Science, Katholieke Universiteit Leuven, 3000, Belgium
[2] IRLab, Computer Science Department, A Coruña University, Spain
christina.lioma@cs.kuleuven.be, rblanco@udc.es,
sien.moens@cs.kuleuven.be

Abstract. Query Expansion (QE) refers to the Information Retrieval (IR) technique of adding assumed relevant terms to a query in order to render it more informative, and hence more likely to retrieve relevant documents. A key problem is how to identify the terms to be added, and how to integrate them into the original query. We address this problem by using as expansion terms social tags that are freely available on the Web. We integrate these tags into the query by treating the QE process as a logical inference (initially proposed in [3]) and by considering the addition of tags as an extra deduction to this process. This work extends Nie's logical inference formalisation of QE to process social tags, and proposes an estimation of tag salience, which is experimentally shown to yield competitive retrieval performance.

1 Introduction

Query Expansion (QE) is an Information Retrieval (IR) technique that aims to expand queries with assumed relevant terms in order to render them more informative and hence facilitate the retrieval of relevant documents. Typically, the terms used for expansion are fetched from some collection or thesaurus, and weighted in different ways. There exists a variety of different ways to do so, overviewed in [5]. We present an approach to QE that uses social tags to expand queries. We collect these tags from the Web, and estimate their salience before using them to expand user queries. We formalise this as a logical inference approach, following Nie's original such formalisation of QE [3]. We extend Nie's logical inference representation by adding an additional estimation of the content salience of social tags. This work contributes an illustration of the ease with which Nie's treatment of QE as logical inference can be extended to accommodate further sources of QE, social tags in this case. In addition, this work contributes a novel estimation of tag salience. Illustrative experiments show our proposed tag-QE approach to yield competitive retrieval performance.

Section 2 presents the logical inference approach to QE with social tags, Section 3 presents illustrative experimental results, and Section 4 summarises this work and outlines future research directions.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 358–361, 2009.

2 Query Expansion with Social Tags as Logical Inference

Let K represent a knowledge system upon which all inference is made. Let d denote a document, and q denote a query. Then, the relevance of d to q with respect to this system can be expressed as $K \vdash d \rightarrow q$. If one can prove that $K \vdash d \rightarrow q$, then the document is said to be relevant to the query, otherwise the document is said to be irrelevant to the query. Nie [3] applies this representation to model QE, by defining a new query q' that constitutes an expanded expression of the original query q. Then, by applying classical logic transitivity, the evaluation of $K \vdash d \rightarrow q$ can be done as follows (K is removed henceforth): $d \rightarrow q' \wedge q' \rightarrow q \vdash d \rightarrow q$. This relation means that the new query q' is satisfied (implied) by the document, in which case the original query q is also satisfied by the document. Because q' can be any query expression, the above deduction can be written as: $\vee_{q'}(d \rightarrow q' \wedge q' \rightarrow q) \vdash d \rightarrow q$. Interpreting this formula in a context that involves uncertainty, the following function P can be defined:

$$P(d \rightarrow q) = P(\vee_{q'}(d \rightarrow q' \wedge q' \rightarrow q)) \tag{1}$$

where $P(d \rightarrow q')$ measures the degree of direct satisfaction of query q' to document d, and $P(q' \rightarrow q)$ measures the degree of relatedness of query q' to the original query q. Eq. 1 can be interpreted as the probability $P(R|q,d)$ that a document d is relevant to a query q as follows: $P(R|d,q) = \sum_{q'} P(R,q'|d,q) = \sum_{q'} P(R|d,q,q')P(q'|d,q)$. Assuming that q' is a good approximation of q leads to: $P(R|d,q,q') = P(R|d,q')$. The derivation of q' depends only on q, not on d, hence $P(q'|d,q) = P(q'|q)$. Based on this, we get the following expression:

$$P(R|d,q) = \sum_{q'} P(R|d,q')P(q'|q) \tag{2}$$

where $P(R|d,q')$ denotes the relevance estimation of the document to the derived query, and $P(q'|q)$ denotes the relationship between the original query q and the derived query q'. Eq. 2 can be rewritten in order to express QE on the basis of individual terms, rather than whole queries, as follows (see [3] for the full derivation):

$$P(R|d,q) = \sum_{t'} P(R|d,t')P(t'|q) \tag{3}$$

where t' denotes a term in the expanded query. This formula allows us to consider the uncertainty of the correspondence between the expansion terms and the original query terms as a factor in the estimation of relevance.

Eq. 3 has two components. The first component, $P(R|d,t')$, may be interpreted as the term weight within a document, and can be estimated by various different ranking models, for instance with Okapi's BM25 [4], which we use in this work. The second component, $P(t'|q)$, may be interpreted as the term importance of a query, and has to be estimated in a way that reflects the probability of finding an expansion term in the query. Applied to our case of QE with tags, $P(t'|q)$ denotes the probability of finding a tag (denoted τ) in the query. This probability must be estimated in a way that reflects the salience of the tag. We propose the following IDF-like approximation:

$$P(\tau|q) = \frac{N}{n_\tau} \tag{4}$$

where N is the number of documents in the collection, and n_τ is the number of documents in the collection that contain the tag τ. The aim of Eq. 4 is to discriminate between tags on the basis of how many documents within a large collection are associated to them (hence 'tagged' by them). Eq. 4 is one suggestion for estimating tag salience, which we evaluate experimentally in Section 3. Further alternative estimations are possible, for instance by relatively straight-forward extensions to IDF, such as RIDF [2], or by more elaborate approximations of tag topicality, such as Zhou et al.'s approach [6] that uses Bayesian Inference.

3 Experimental Evaluation

We present an illustrative evaluation of our proposed QE with tags, which is organised as follows: The baseline is standard retrieval without QE. This baseline is compared against our proposed QE with tags. In order to contextualise this comparison, we further compare these results to a state-of-the-art retrieval with conventional QE (i.e. QE that uses weighted terms for expansion). At all times, retrieval is realised with BM25, whose parameter b is tuned separately for Mean Average Precision (MAP) and Precision at 10 (P10). For conventional QE with terms, we use DFR's Bo1 model [1]. Both conventional QE and our proposed QE include as parameters (i) the number of terms (resp. tags) used for expansion, and (ii) the number of documents from which these terms (resp. tags) are drawn. We tune these parameters by varying them between 1-30 (for terms or tags) and 1-10 (for documents) separately for MAP and P10. Finally, the tags used for our proposed QE are collected by querying Del.icio.us, similarly to [6], whereas the terms used for conventional QE are collected from the same collection used for retrieval. The retrieval collection is the TREC BLOG06 collection (25GB) with queries 901-950 (title only). For our QE with tags, when applying Eq. 4, we compute N and n_τ from the BLOG06 collection, because we do not have access to the statistics of the collection used by Del.icio.us. We assume that Del.icio.us uses a very large collection, and that BLOG06 is a large enough approximation of it (in terms of size). Table 1 displays the performance of our retrieval experiments without QE, with our proposed tag-QE, and with conventional term-QE. We see that our proposed QE outperforms both the baseline and the conventional term-QE at all times, and with respect to both mean and early precision. This observation indicates that our use of tags enhances retrieval performance, not only by fetching a bigger number of relevant documents, but also by fetching more precise documents (i.e. documents of higher relevance to the query). This observation may indicate that the IDF-like formula we proposed to estimate tag salience was a successful approximation, an indication worth analysing further. Overall, these experiments indicate that social tags, when filtered appropriately, may benefit IR, a conclusion also echoed by Zhou et al. [6].

Table 1. Mean Average Precision (MAP) and Precision at 10 (P10) using as baseline BM25 without QE. Against this baseline we compare our proposed method of QE with Del.icio.us tags. To contextualise this comparison, we also display conventional QE with terms. Δ marks the % difference from the baseline.

measure	**BM25 - No QE**	**BM25+QE tags**	Δ%	**BM25+QE terms**	Δ%
MAP	0.3517	0.3636	+3.38	0.3519	+0.06
P10	0.6220	0.6540	+5.14	0.6420	+3.21

4 Conclusion

We presented an approach to Query Expansion (QE) that adds to a query tags collected from a free online social tagging system. By formalising QE as a logical inference process, as proposed by [3], we were able to integrate into ranking an approximation of the uncertainty that a tag is relevant to the original query terms. Specifically, we realised this approximation by proposing an IDF-like weight of tag salience, which considers how many documents in a collection are tagged by a given tag. Both the treatment of QE as logical inference, and the proposed weight of tag salience used clean and tractable estimations. An illustrative experimental evaluation with a 25GB TREC collection showed our proposed tag-QE technique to outperform a baseline of no QE, as well as a state-of-the-art QE model that uses weighted terms for expansion. This is a first positive indication that social tags, when filtered appropriately, may benefit IR. Future research will be geared toward refining the estimation of tag salience, and analysing in depth the effect of tag-QE on a per query basis.

Acknowledgements. Author 2 was partially supported by *Ministerio de Ciencia e Innovación*, FEDER and *Xunta de Galicia* under projects TIN2008-06566-C04-04 and 07SIN005206PR.

References

1. Amati, G.: Probabilistic Models for Information Retrieval based on Divergence from Randomness. PhD thesis, University of Glasgow (2003)
2. Church, K.W., Gale, W.A.: Poisson mixtures. Natural Language Engineering 1(2), 163–190 (1995)
3. Nie, J.-Y.: Query expansion and query translation as logical inference. JASIST 54(4), 335–346 (2003)
4. Robertson, S., Walker, S., Beaulieu, M., Gatford, M., Payne, A.: Okapi at TREC-4. In: Harman, D.K. (ed.) NIST Special Publication 500-236: TREC-4, pp. 73–96. Springer, Heidelberg (1995)
5. Xu, J., Croft, W.B.: Query expansion using local and global document analysis. In: SIGIR, pp. 4–11 (1996)
6. Zhou, D., Bian, J., Zheng, S., Zha, H., Lee Giles, C.: Exploring social annotations for information retrieval. In: WWW, pp. 715–724 (2008)

Evaluating Mobile Proactive Context-Aware Retrieval: An Incremental Benchmark

Davide Menegon[1], Stefano Mizzaro[1], Elena Nazzi[2], and Luca Vassena[1]

[1] University of Udine,
Via delle Scienze, 208, Udine, Italy
{menegon,mizzaro,vassena}@dimi.uniud.it
[2] IT-University of Copenhagen,
Rued Langgaards Vej 7, DK-2300 Copenhagen S
elna@itu.dk

Abstract. We present the evaluation of a novel application for Web content perusal by means of context-aware mobile devices that proactively query an external search engine. To this aim, we develop a TREC-like benchmark and we use it to evaluate different strategies for automatic query construction on the basis of user's current context. We discuss both the methodology and the results.

Keywords: TREC, IR effectiveness, location based systems.

1 Introduction

This work is in the Context-Aware Retrieval (CAR) field [1,2], and it is centered on the Context-Aware Browser (CAB) [3], a new approach to proactive context-aware Web content perusal by means of mobile devices (cellphones, PDAs, etc.). The main idea behind CAB is to empower a generic mobile device with a browser able to automatically and dynamically retrieve and load Web pages, services, and applications according to user's current context (roughly described as the situation the user is in). The context is represented by a set of terms, automatically inferred by means of a Bayesian network on the basis of data received from sensors. In the retrieval process, starting from the context representation, a query is automatically built and sent to an external Web search engine (we use Yahoo! APIs http://developer.yahoo.com/search/web/), in order to find the most suitable Web pages for the sensed context. CAB allows a "physical browsing": browsing the digital world based on the situations in the real world.

CAB development is underway: most of its components have been implemented, but the retrieval mechanism can exploit several different strategies, that must be compared. With this aim, we propose a TREC-like evaluation benchmark, discuss its limits, and test it. Although we focus on CAB, the problems we try to solve are typical of any proactive CAR system. Indeed, this work extends [4]: we now concentrate on the retrieval of Web pages (in place of applications and their manually created descriptors) and we adopt an incremental and dynamic benchmark, as explained below.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 362–365, 2009.

```
<contextDescriptor>
<title> Heathrow airport </title>
<description>
The user has just landed at London Heathrow international airport. He is looking at a flight
timetable and at a timetable for connections to London. It is lunch time.
</description>
<narrative> ...</narrative>
<relevance>
A Web page is relevant: it contains information about a flight, about the means of transport
to reach town, about bars and fastfoods in the airport, or it allows to book a flight. A
web page that contains only one of these aspects is relevant; if it contains some links to
relevant pages is partially relevant. If the judge is not able, for any reason, to judge the
page, its value is ‘‘I don’t know’’.
</relevance>
...
</contextDescriptor>
```

Fig. 1. A (part of a) context descriptor

2 CREC Benchmark

The CREC (*CAB Retrieval Evaluation Collection)* benchmark is constituted by the usual three components: topics, document collection, and relevance judgments. Topics (information need descriptions) are context descriptors (see Fig. 1), which represent different user's contexts in different domains, and have been designed similarly to TREC topics; CREC includes 10 context descriptors which differ for user activities, location, time, etc. The relevance judgments have been made by a unique judge using a four level relevance scale.

The collection consists of Web pages, and it is dynamic, i.e., it evolves during the tests. We built two CREC versions so far. The first version has been constructed performing 5 manual queries for each topic, and judging the first 150 single retrieved documents for each topic. Starting from this version of the collection, we adopted an "interactive search and judge" [5] approach to add more relevant documents. In particular we ran some queries, automatically built from context descriptors and, for each query, the Web pages that were not already in the collection and were retrieved in the first 10 ranks have been added to the collection and judged, obtaining its second version (3634 total pages: 494 relevant, 596 partially relevant, 34 not classified, and 2510 not relevant).

CREC is not static because, if a new implementation of the CAB external search engine needs to be evaluated, CREC will not contain, in general, all the retrieved pages (a new strategy might obviously retrieve new pages, and the Web is dynamic). Since this would make the evaluation less reliable, the collection will be extended by including the newly retrieved documents, and judging them.

3 Evaluation

We used CREC to compare four automatic query construction strategies. All of them work on term lists automatically extracted from the `<description>` field: for instance, the context descriptor in Fig. 1 is seen by the strategies as "user just landed london heathrow international airport looking flight timetable timetable

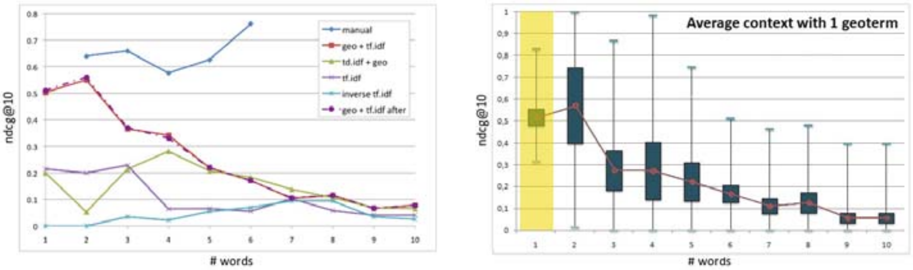


Fig. 2. Results: the five strategies (left), and geo+tf.idf details (right)

connections london lunch time" (conversely, the human relevance judge uses the whole context descriptor). The strategies are based on two main indexes: *tf.idf* and *geoterms* (i.e., terms that refer to geographical location information — in our approach, a term is a geoterm if its Wikipedia page contains geographical coordinates). We chose tf.idf as it is a classical and largely used IR technique, and geoterms because location is probably the contextual dimension that is more informative of user's current context. These indexes, differently combined, are used to rank the term lists according to their importance (that will be different for the four strategies). For each strategy and context descriptor, 10 queries of different lengths (from 1 to 10 terms) are automatically formulated, incrementally selecting the first 10 terms of the ranked lists. Thus query construction is incremental: once a term is in a query, it will remain in longer queries as well. We also use, as an upper reference strategy, the manual approach, where a mobile user directly chooses terms and defines her query (we used the queries generated to build the first version of the document collection, see Sect. 2).

We measure strategies effectiveness by means of nDCG@10: since it is unlikely that CAB users will scroll long lists of retrieved items, it is reasonable to consider only the first 10 retrieved items.

4 Results

Fig. 2 (left) compares the four strategies, and the manual one, showing their effectiveness (nDCG@10, on the Y axis) averaged on all 10 contexts, for different query lengths (X axis). Apart from the manual one, the most effective strategy is the *geo+tf.idf*. In this strategy, first all the geoterms are added to the query, then the other terms, ranked by decreasing tf.idf, follow. Fig. 2 (right) shows, besides average, also min, max, and variance. Further analysis of the data, not reported here for brevity, shows that the maximum performance is obtained when just one tf.idf term is added after all the geoterms (each context contains 1, 2, or 3 geoterms), then nDCG@10 decreases. Long queries have low performance.

All the proposed strategies have lower performance than the manual one — see the higher curve in Fig. 2 (left) —, therefore they can be improved. Moreover, the manual strategy tends to become more effective with longer queries: one reason is that automatic strategies are constrained by the incremental query construction (see Sect. 3), whereas manual strategy is not.

As relevance judgments have been made by a unique judge, to verify that subjectivity is not an issue for our benchmark, we performed an additional experiment, involving two more judges. Measuring inter-judge agreement on a pool of retrieved pages, judgments on average agreed on 65% pages (which became 92% after a discussion between judges). New relevant pages retrieved by the two new judges were also added to the collection, and the evaluation performed again for geo+tf.idf. The dashed line in Fig. 2 (left) shows the effectiveness of geo+tf.idf computed considering the new pages and judgments: it does not change significantly. Thus CREC seems reliable, at least to a reasonable extent.

To conclude, the CREC benchmark helped the development process giving good insights (e.g., high effectiveness is obtained by adding just one term after the geoterms) and underlining weak points (e.g., adding more and more terms in the query does not increase effectiveness). Also, once the benchmark is configured, it can be reused to test new strategies or related features, in a semi-automatic way; new judgments are needed, but the effort is lower than that required for a user study (judging time for the 2nd version was 37 hours). Thus, our approach seems adequate for early stage evaluations, useful to better inform the following more focussed, and demanding, user studies.

In the future we will work on two issues: to seek for more effective strategies that better compete with the manual one (e.g., by removing the constraint of incremental query construction) and, more generally, to better understand reliability and usefulness of our incremental benchmark approach. In general, our approach seems interesting and valid for IR applications that need high precision, like CAB (for which nDCG@10 is an adequate metric); with high recall, the effort for the new judgments would probably be too high.

References

1. Göker, A., Myrhaug, H.: Evaluation of a mobile information system in context. Information Processing & Management 44(1), 39–65 (2008)
2. Jones, G.J.F., Brown, P.J.: Context-aware retrieval for ubiquitous computing environments. In: Crestani, F., Dunlop, M.D., Mizzaro, S. (eds.) Mobile HCI International Workshop 2003. LNCS, vol. 2954, pp. 227–243. Springer, Heidelberg (2004)
3. Coppola, P., Della Mea, V., Di Gaspero, L., Mischis, D., Mizzaro, S., Scagnetto, I., Vassena, L.: AI techniques in a context-aware ubiquitous environment. In: Ella Hassanien, A., Abraham, A., Hagras, H. (eds.) Pervasive Computing: Innovations in Intelligent Multimedia and Applications. Computer Communications and Networks. Springer, Heidelberg (2009)
4. Mizzaro, S., Nazzi, E., Vassena, L.: Retrieval of context-aware applications on mobile devices: how to evaluate? In: Proc. of Information Interaction in Context (IIiX 2008), pp. 65–71 (2008)
5. Cormack, G.V., Palmer, C.R., Clarke, C.L.A.: Efficient construction of large test collections. In: Proc. of SIGIR 1998, pp. 282–289. ACM, New York (1998)

Predicting the Usefulness of Collection Enrichment for Enterprise Search

Jie Peng, Ben He, and Iadh Ounis

Department of Computing Science,
University of Glasgow, G12 8QQ, UK
{pj,ben,ounis}@dcs.gla.ac.uk

Abstract. Query Expansion (QE) often improves the retrieval performance of an Information Retrieval (IR) system. However, as enterprise intranets are often sparse in nature, with limited use of alternative lexical representations between authors, it can be advantageous to use Collection Enrichment (CE) to gather higher quality pseudo-feedback documents. In this paper, we propose the use of query performance predictors to selectively apply CE on a per-query basis. We thoroughly evaluate our approach on the CERC standard test collection and its corresponding topic sets from the TREC 2007 & 2008 Enterprise track document search tasks. We experiment with 3 different external resources and 3 different query performance predictors. Our experimental results demonstrate that our proposed approach leads to a significant improvement in retrieval performance.

1 Introduction

Collections within enterprises are often characterised by their limited vocabulary, since they are written by a small number of people, following specific guidelines and aims. Therefore, while query expansion is usually effective in IR, the limited use of alternative lexical representations within enterprise collections could lead to poor pseudo-relevance sets. In this case, it seems intuitive to make use of the well-established Collection Enrichment (CE) technique, which performs query expansion on a larger and higher-quality external resource [1,2]. The reformulated query is then used to retrieve documents from the local enterprise collection. However, the quality of the external collection is a key factor that affects the retrieval performance given by CE [2].

In this paper, we argue that the retrieval performance of document search within an enterprise can be further enhanced by applying CE in a selective manner on a per-query basis. The idea is that the usefulness of the local or external collection for QE varies from a query to another. Using query performance predictors, we propose a decision mechanism that indicates the appropriateness of the local and external collections for a given query. QE is then applied on the collection, either local or external, that is predicted to contain higher quality of relevant content for the query.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 366–370, 2009.

To the best of our knowledge, this is the first study that investigates the usefulness of selectively applying CE in an enterprise setting. The proposed selective CE mechanism is thoroughly evaluated on the standard CERC test collection and its corresponding topic sets from TREC Enterprise track 2007 & 2008. We apply three different external resources and three different query performance predictors. Our experimental results show that our selective application of CE provides a significantly better retrieval performance than an approach that systematically applies QE on either the local or external collection.

Table 1. The decision mechanism for the selective application of CE. *local*, *external* and *disabled* in the column *Decision* indicate expanding the initial query on the *local* resource, *external* resource and disabling the expansion, respectively.

$score_L > T$	$score_E > T$	$score_L > score_E$	Decision
True	True or False	True	local
True or False	True	False	external
False	False	True or False	disabled

2 Selective Collection Enrichment

Our decision mechanism enriches the enterprise collection only if the external resource is predicted to contain more relevant content to the query than the local collection. For a given query, we use query performance predictors to estimate the quality of the pseudo-relevance sets returned by either the local or the external collection. A lower score corresponds to a difficult query for that collection [3], while a higher score suggests a richer pseudo-relevance set. Using the query difficulty scores returned by the predictors, our decision mechanism applies QE on the collection that corresponds to the higher predictor score, i.e. the collection that is predicted to lead to a better retrieval performance.

In addition, if the predictor scores on both the local ($score_L$) and the external ($score_E$) collections are lower than a threshold (T), then query expansion is not applied for that given query. Table 1 summarises our proposed decision mechanism for the selective application of collection enrichment.

3 Experimental Setting

Three popular query performance predictors, namely the Average Inverse Collection Term Frequency (AvICTF) and the $\gamma 2$ pre-retrieval predictors [3], and the Clarity Score (CS) post-retrieval predictor [4], are studied in this paper. These predictors have been widely applied in the literature and were shown to be generally effective in predicting query performance. Note that unlike the pre-retrieval predictors, the CS predictor involves a parameter that needs tuning. It is also of note that for the CS predictor, a lower score indicates a better retrieval performance [4]. Therefore, unlike for AvICTF and $\gamma 2$, the decision mechanism is

Table 2. Evaluation of the selective application of CE on the TREC 2008 enterprise document search task

	Wikipedia			Aquaint 2			.GOV		
	MAP	nDCG	Acc.	MAP	nDCG	Acc.	MAP	nDCG	Acc.
PL2F	0.3629	0.5502	-	0.3629	0.5502	-	0.3629	0.5502	-
+QE	*0.3811*	*0.5646*	-	*0.3811*	*0.5646*	-	*0.3811*	*0.5646*	-
+CE	0.3684	0.5606	-	0.3402	0.5391	-	0.3583	0.5551	-
MAX	0.4204 ⋆	0.5978 ⋆	100%	0.4022 ⋆	0.5832 ⋆	100%	0.4135 ⋆	0.5930 ⋆	100%
Selective CE by using Predictors									
AvICTF	0.3660 ↓	0.5567 ↓	40.98%	0.3576 ↓	0.5500 ↓	45.00%	0.3765 ↓	0.5570 ↓	55.73%
$\gamma 2$	**0.3959** ↑ ⋆	**0.5679** ↑	67.21% ∗	**0.3825** ↑	**0.5673** ↑	58.33%	0.3864 ↑	0.5658 ↑	59.01%
CS	0.3824 ↑	0.5653 ↑	49.18%	0.3658 ↓	0.5475 ↓	51.66%	**0.3941** ↑	**0.5782** ↑	67.21% ∗

reversed to favour collection with a lower predictor score. However, the principle of deciding on the use of the collection enrichment is the same.

We use the standard CERC enterprise test collection [5], and its corresponding title-only topics from the TREC Enterprise track 2007 & 2008, respectively. There are 42 and 63 judged topics from TREC 2007 & TREC 2008, respectively. We experiment with three external resources, namely Wikipedia[1], Aquaint 2[2], and the TREC .GOV collection. For indexing and retrieval, we use the Terrier IR platform[3], and apply standard stopword removal and the Porter's stemming algorithm for English. For the CERC and .GOV collections, we index the body, anchor text and titles of the documents as separate fields. For the Wikipedia and the Aquaint 2 collections, we do not use the anchor text field as our initial experiments show that it is not beneficial for retrieval. We use the Bo1 term weighting model for query expansion [6]. Documents are ranked using the PL2F field-based DFR document weighting model [7]. The parameters that are related to the PL2F document weighting model, the CS predictor and the threshold T of the decision mechanism are set by optimising MAP on the TREC 2007 dataset, using a simulated annealing procedure [8]. We evaluate our method using the TREC 2008 topics.

4 Experimental Results

Table 2 presents the evaluation results of our proposed method. As shown in the table, the use of query expansion on the enterprise collection (PL2F+QE) outperforms PL2F, as well as a system that systematically applies QE on the external collection (denoted PL2F+CE) across three different external resources. Hence, in order to compare our proposed method with a strong baseline, we use PL2F + QE as our baseline. ↑ & ↓ denote that the obtained retrieval performance by using the predictor is better (resp. worse) than the baseline. The *Acc.* column shows the accuracy of our proposed method, which is given by the number

[1] http://en.wikipedia.org/wiki/Wikipedia:Database_download

[2] http://trec.nist.gov/act_part/tracks/qa/qa.07.guidelines.html

[3] http://terrier.org

of queries that has been appropriately applied with CE divided by the total number of queries. The symbol $*$ denotes that the predictor makes a correct prediction for a statistically significant number of queries, according to the Sign Test ($p < 0.05$). Values that are statistically better than the baseline are marked with $\star$ (Wilcoxon Matched-Pairs Signed-Ranks Test, $p < 0.05$).

Firstly, we assess how important it is to selectively apply CE on a per-query basis, by estimating the MAP and nDCG upper bounds (highlighted with underline). In this case, if query expansion is deemed helpful for the query, we manually select the most appropriate collection, from which to build the pseudo-relevance set. From Table 2, it is clear that the retrieval performance with the manual selective application of CE leads to a significant improvement over the systems that apply systematically either QE or CE, across three different external resources. This suggests that identifying the most appropriate collection for query expansion on a per-query basis is useful.

Secondly, we test how effective our proposed selective application technique is, by comparing the performance obtained by using the query performance predictor to the baseline that systematically applies QE on the local collection (PL2F+QE). From Table 2, we can see that the best retrieval performance (excluding the upper bounds), highlighted in bold, in each column is obtained by using our proposed method. We also observe that, in some cases, a statistically significant number of queries have been correctly applied with CE. In particular, the $\gamma 2$ predictor has led to a significant improvement in MAP. This suggests that our proposed approach is an effective method for selectively applying CE.

Finally, we investigate the importance of the choice of external resources and predictors. From Table 2, we can see that the systematic application of CE can harm the retrieval performance (e.g. Aquaint 2), compared to the results obtained by using PL2F only; while the $AvICTF$ predictor constantly decreases the retrieval performance. This suggests that the choice of an appropriate external resource and predictor before the application of selective CE is very important. In addition, we find that the $\gamma 2$ predictor constantly enhances the retrieval performance across 3 different external resources. In fact, the highest MAP score is achieved by using the $\gamma 2$ predictor on the Wikipedia collection. Besides, the $\gamma 2$ predictor is parameter-free and only relies on the statistics of the collection.

5 Conclusions

We have proposed the use of query performance predictors to selectively apply CE on a per-query basis for document search within an enterprise. The experimental results show that the retrieval performance can be significantly improved when the external resource and the predictor have been appropriately chosen. In particular, the $\gamma 2$ predictor is the most efficient, effective and robust predictor for the enterprise document search. In the future, we plan to deploy our proposed method for blog search as collections from the blogosphere contain many spam documents and other noisy vocabulary, meaning that query expansion might benefit from the use of high-quality external resources.

References

1. Diaz, F., Metzler, D.: Improving the Estimation of Relevance Models using Large External Corpora. In: Proceedings of SIGIR 2006 (2006)
2. Kwok, K.L., Chan, M.: Improving two-stage ad-hoc retrieval for short queries. In: Proceedings of SIGIR 1998 (1998)
3. He, B., Ounis, I.: Query Performance Prediction. In: Information Systems (2004)
4. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting Query Performance. In: Proceedings of SIGIR 2002 (2002)
5. Bailey, P., Craswell, N., de Vries, A.P., Soboroff, I.: Overview of the TREC 2007 Enterprise Track. In: Proceedings of TREC 2007 (2007)
6. Amati, G.: Probabilistic Models for Information Retrieval based on Divergence from Randomness. PhD thesis, UK (2003)
7. Macdonald, C., Plachouras, V., He, B., Lioma, C., Ounis, I.: University of Glasgow at WebCLEF 2005: Experiments in Per-field Normalisation and Language Specific Stemming. In: Peters, C., Gey, F.C., Gonzalo, J., Müller, H., Jones, G.J.F., Kluck, M., Magnini, B., de Rijke, M., Giampiccolo, D. (eds.) CLEF 2005. LNCS, vol. 4022, pp. 898–907. Springer, Heidelberg (2006)
8. Kirkpatrick, S., Gelatt, C., Vecchi, M.: Optimization by simulated annealing. Science 220(4598) (1983)

Ranking List Dispersion as a Query Performance Predictor*

Joaquín Pérez-Iglesias and Lourdes Araujo

Universidad Nacional de Educación a Distancia
Madrid 28040, Spain
joaquin.perez@lsi.uned.es, lurdes@lsi.uned.es

Abstract. In this paper we introduce a novel approach for query performance prediction based on ranking list scores dispersion. Starting from the hypothesis that different score distributions appear for good and poor performance queries, we introduce a set of measures that capture these differences between both types of distributions. The use of measures based on standard deviation of ranking list scores, as a prediction value, shows a significant correlation degree in terms of average precision.

1 Introduction

During the last years a growing attention has been focused on the problem of query performance prediction. This topic has turned into an important challenge for the IR community. Query performance prediction deals with the problem of detecting those queries for which a search system would be able to return a document set useful for an user. The proposed method for query performance prediction falls into post-retrieval prediction methods. This type of predictors make use of the information supplied from the search system once the search has been carried out. This work is based on the hypothesis that different scores distributions for good and poor performance queries can be observed.

Related approaches that use ranking list scores can be found in the works carried out by Diaz [1], where the similarity between the scores of topically close documents, is applied as a prediction value. A similar approach was proposed by Vinay [2], in this case the prediction is based on the correlation between the actual rank and a computed expected rank, where the expected rank is obtained modelling the score of a document as a Gaussian random variable.

2 Ranking List Scores Dispersion as a Predictor

The approach proposed on this paper is based on the study of the ranking list obtained after a retrieval process is executed. A search system ranks the related

* This work has been partially supported by the Regional Government of Madrid under the Research Network MAVIR (S-0505/TIC-0267) and the Spanish MICINN project TIN2007-68083.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 371–374, 2009.

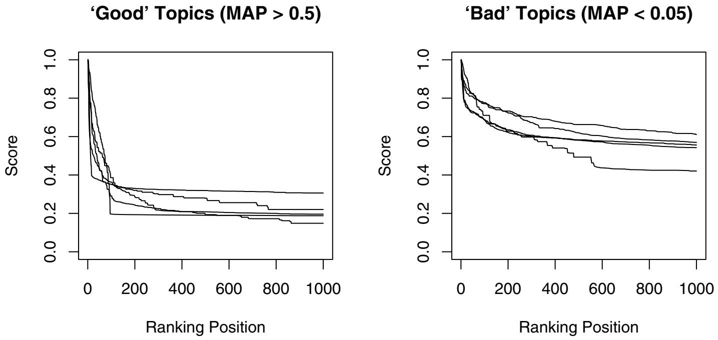


Fig. 1. 5 Best Performing Topics (left) Vs 5 Worst Performing Topics (right), from Robust 2004 using BM25. Scores have been normalised in $[0, 1]$. The maximum number of retrieved documents has been fixed to 1000.

documents found within the collection. For this purpose a ranking function assigns a weight (or score) to each document in the collection. In a 'naive' sense the scores can be interpreted as 'quantitative measures' of the documents relevance. The ranking list scores distribution can be an indicative of the quality performance for a specific topic. Based on this premise some differences between document scores distribution, for good and poor performing topics should be observed.

For example, if a ranking list has a high value of dispersion among the document scores, it could be a sign that the ranking function has been able to discriminate between relevant and not relevant documents. On the other hand if a low level of dispersion appears, because the ranking function has assigned similar weights, it can be interpreted as it was not able to distinguish between relevant and not relevant documents.

Differences in terms of scores dispersion can be observed in figure 1 for the topics that achieve the best performance and those that obtain the lowest values in terms of AP (Average Precision) for Robust 2004 [3].

2.1 Proposed Measures

In this work we have tested different approaches to capture and measure dispersion along the obtained ranking list. Some prior studies have tried to model how document weights are distributed along a ranking list. In general, it can be assumed that an adequate model could be a mix between an exponential and a normal probability distribution. Exponential for not relevant documents, and normal for relevant documents [4,5]. Generally a majority of retrieved documents are not relevant (exponential distribution), thus it is likely that a great number of documents will be weighted with a low score. As a consequence, a ranking list shape holds a long *tail* where a majority of not relevant documents are placed.

Some notation is needed to define the next measures: (i) A ranking list RL is a document list sorted in decreasing order by their documents scores; (ii) The score assigned to a document, placed at position i into the ranking list, is defined as $score(d_i)$.

Standard Deviation: Given ranking list scores mean $\mu(RL)$, standard deviation is computed as next:

$$\sigma(RL) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(score(d_i) - \mu(RL))^2} \tag{1}$$

A drawback in the use of the standard deviation is caused by the great number of low scores assigned by the ranking function. As was described previously, a high percentage of document scores have a low value, which causes that mean is displaced towards the region of densest distribution, that is the tail of the ranking list. As a consequence of it, the deviation on the top documents is not captured properly when the standard deviation is computed along the full ranking list.

Maximum Standard Deviation: In order to minimise the effect of low scores high frequency, the maximum standard deviation is proposed. This estimator is based on the idea of computing the standard deviation at each point in the ranking list, and selecting the maximum value.

$$\sigma_{max} = max[\forall d \in RL, \sigma(RL_{[1,d]}))] \tag{2}$$

Standard Deviation at *k*: Standard deviation measured at a cut point k of the ranking list (σ_k). With the selection of a suitable k value, the noise introduced by low scores is removed. The k value is fixed at the ranking position that maximise the correlation degree with AP.

3 Results and Conclusions

The different measures proposed in this paper has been tested with the set of documents from TREC Disk4 & 5, minus Congressional Record and the topics used in the Robust 2004 track[1]. Only the field title from topics has been employed in the experiments. We have selected three well-known retrieval models (BM25,LM and PL2) to test the validity and compare the obtained prediction values among them.

The obtained results[2] appears in table 1. These experiments were executed with a default ranking list size of 1000, this was the default number of documents employed for the calculation of MAP in Robust 2004.

As can be seen the obtained correlation coefficients, with the same measure, for different retrieval models are similar. As it was expected a common behaviour for the proposed retrieval models can be observed.

[1] Topic 672 has been removed since no relevant documents can be found for it in the collection.

[2] The correlation coefficients obtained are statistically significant at a level of 0.01.

In relation with the ability of the proposed measures to capture dispersion, the best results have been obtained with the selection of an optimal ranking list size k for σ_k. The size of the ranking list that maximises the correlation for all retrieval models is 100. Opposite to this, standard deviation exhibits a worse performance than the rest of measures as was affected by the described problem of the *ranking list tail.* On the other hand the results obtained with the maximum standard deviation outperforms to those achieved with standard deviation. Therefore σ_{max} avoids, at least in part, the lack of precision, in terms of dispersion measurement, obtained by the classic standard deviation.

Table 1. Pearson and Kendall correlation coefficients obtained with the proposed measures for different retrieval models. Strongest correlation values appear in bold.

	BM25		LM		PL2	
	Pearson	Kendall	Pearson	Kendall	Pearson	Kendall
σ	0.39	0.34	0.35	0.33	0.30	0.29
σ_{max}	0.40	0.41	0.40	0.39	0.37	0.37
σ_{100}	**0.55**	**0.41**	**0.53**	**0.41**	**0.53**	**0.39**

The obtained results show that measures based on standard deviation over scores ranking list, can be used to predict the quality of a search system reply. Further research in the selection of a suitable cut point for measuring the standard deviation, should be carried out to improve the obtained results.

References

1. Diaz, F.: Performance prediction using spatial autocorrelation. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR 2007, p. 583. ACM Press, New York (2007)
2. Vinay, V., Milic-Frayling, N., Cox, I.: Estimating retrieval effectiveness using rank distributions. In: CIKM 2008: Proceeding of the 17th ACM conference on Information and knowledge management, pp. 1425–1426. ACM, New York (2008)
3. Voorhees, E.M.: Overview of the trec 2004 robust retrieval track. In: Proceedings of the Thirteenth Text REtrieval Conference, TREC (2004)
4. Manmatha, R., Rath, T., Feng, F.: Modeling score distributions for combining the outputs of search engines. In: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR 2001, pp. 267–275 (2001)
5. Robertson, S.: On score distributions and relevance. Advances in Information Retrieval 4425, 40–51 (2007)

Semi-subsumed Events: A Probabilistic Semantics of the BM25 Term Frequency Quantification

Hengzhi Wu and Thomas Roelleke

Queen Mary, University of London
{hzwoo,thor}@dcs.qmul.ac.uk

Abstract. Through BM25, the asymptotic term frequency quantification $\mathrm{TF} = \mathrm{tf}/(\mathrm{tf}+K)$, where tf is the within-document term frequency and K is a normalisation factor, became popular. This paper reports a finding regarding the meaning of the TF quantification: in the triangle of independence and subsumption, the TF quantification forms the altitude, that is, the middle between independent and subsumed events. We refer to this new assumption as semi-subsumed. While this finding of a well-defined probabilistic assumption solves the probabilistic interpretation of the BM25 TF quantification, it is also of wider impact regarding probability theory.

1 Introduction and Motivation

The BM25 TF quantification/normalisation of the form $\mathrm{tf}/(\mathrm{tf}+K)$ where tf is the total within-document frequency and K is a normalisation parameter (includes the pivoted document length) is renown for superior retrieval quality, outperforming by far the bare total count tf or a maximum-likelihood estimate of the form tf/N_d where N_d is the document length. The total tf corresponds to an *independence* assumption. That is each occurrence of the same term is treated as independent. This assumption is wrong, as the success of BM25 proves.

If a term occurs in a document, and let the initial probability for this occurrence be $P(t|c) = 1/100$ (for example, t occurs in 1% of the documents), then the probability that it occurs again further on in the same document is greater than the initial probability. In other words, the occurrence of an event depends on previous occurrences.

The core contribution of this paper is the notion "semi-subsumed", a probabilistic assumption precisely half-way between independent and subsumed. Probabilistic assumptions are essential in large-scale applications of probabilistic reasoning. Often, the classical assumptions disjointness, independence or subsumption are assumed for events since otherwise the probabilistic reasoning is computationally too expensive. In this paper we focus mainly on the theory around semi-subsumed events, and the effect and application of assuming events to be semi-subsumed in more general probabilistic frameworks such as probabilistic inference networks (PIN) is topic of future research.

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 375–379, 2009.

2 TF-IDF and BM25

The BM25 TF quantification can be viewed as an approximation of the 2-Poisson model ([2]); this is a probabilistic semantics, however, this paper contributes what can be seen as an intuitive assumption.

This section reviews the probabilistic interpretation of TF-IDF. Let $\mathrm{tf} := n_L(t,d)$ denote the within-document term frequency, i.e. the number of *locations* at which term t occurs in document d; similarly, let $\mathrm{df}(t,c) := n_D(t,c)$ denote the number of *documents* containing term t in collection c; $N_D(c)$ is the total number of documents. The notation allows for a consistent representation of the dimensions used in document retrieval ([4]). Then, TF-IDF and BM25 are defined as follows:

$$P_D(t|c) := n_D(t,c)/N_D(c) \tag{1}$$

$$\mathrm{idf}(t,c) := -\log P_D(t|c) \tag{2}$$

$$\mathrm{RSV}_{\text{TF-IDF}}(d,q,c) := \sum_{t \in d \cap q} \mathrm{tf}(t,d) \cdot \mathrm{idf}(t,c) \tag{3}$$

$$\mathrm{RSV}_{\text{BM25}}(d,q,r,\bar{r}) := \sum_{t \in d \cap q} \frac{\mathrm{tf}(t,d)}{\mathrm{tf}(t,d)+K} \cdot w_t \tag{4}$$

$P_D(t|c)$ is the *document-based* term probability, and $\mathit{idf}(t,c)$ is the negative logarithm of this probability. The term weight w_t is the binary independence weight (based on the probabilities $P(t|r)$ and $P(t|\bar{r})$ that t occurs in relevant and non-relevant documents). The $\mathrm{idf}(t,c)$ can be viewed as an approximation of $w_t = -\log 1/P(t|\bar{r})$ for missing relevance, and this constitutes the close relationship of TF-IDF and BM25 ([1,3]).

To demonstrate how TF-IDF/BM25 relate to $P(d|q)$ and an assumption for subsequent term events, the next equation forms the exponent of $\mathrm{RSV}_{\text{TF-IDF}}$.

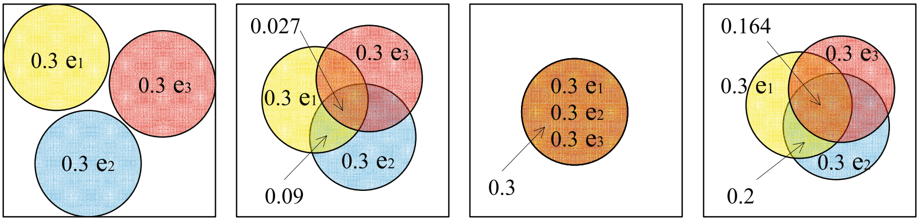
$$\exp(\mathrm{RSV}_{\text{TF-IDF}}) = \prod_{t \in d \cap q} \left(\frac{1}{P_D(t|c)}\right)^{\mathrm{tf}(t,d)} \tag{5}$$

This transformation shows that "naive" TF-IDF involves the expression $P_D(t|c)^{\mathrm{tf}(t,d)}$. $P_D(t|c)$ is the document-based term probability, and the exponent means that "naive" TF-IDF assumes the occurrences of t to be independent events.

The BM25 TF component can be viewed as proposing $P_D(t|c)^{\mathrm{tf}/(\mathrm{tf}+K)}$ to be the term probability, and this probability is significantly greater than $P_D(t|c)^{\mathrm{tf}}$, i.e. the BM25 suggestion is that the probability of subsequent term occurrences is greater than the probability for independent occurrences. The next section shows that this corresponds to assuming subsequent occurrences of a term to be *semi-subsumed* events.

3 Semi-subsumed Events

Figure 1 illustrate the assumption "semi-subsumed" for three occurrences of an event. Semi-subsumed events overlap more than independent events do, but the overlap is less than for fully subsumed events. For example, given the single event probability



(a) Disjoint (b) Independent (c) Subsumed (d) Semi-Subsumed

Fig. 1. Probabilistic assumptions: three event occurrences

	independent occurrences						semi-subsumed				subsumed occurrences	
1							$\frac{1}{1}$					
2						$\frac{2}{1}$	$\frac{2}{3/2}$	$\frac{2}{2}$				
3				$\frac{3}{1}$			$\frac{3}{4/2}$		$\frac{3}{3}$			
4			$\frac{4}{1}$			$\frac{4}{2}$		$\frac{4}{3}$		$\frac{4}{4}$		
5		$\frac{5}{1}$		$\frac{5}{2}$			$\frac{5}{6/2}$		$\frac{5}{4}$		$\frac{5}{5}$	
...					...		...		...			
n	$\frac{n}{1}$	$\frac{n}{2}$		$\frac{n}{3}$			$\frac{n}{(n+1)/2}$		$\frac{n}{n-2}$		$\frac{n}{n-1}$	$\frac{n}{n}$

Fig. 2. Independence-Subsumption Triangle (IST)

$P(e) = 0.3$, for independent occurrences, $P(e_1 \wedge e_2) = 0.3^2 = 0.09$, whereas for subsumed occurrences $P(e_1 \wedge e_2) = 0.3^{2 \cdot 2/3}$.

The independence-subsumption triangle in Figure 2 shows the justification and meaning of the exponent for semi-subsumed events. The left edge of the triangle corresponds to independence, i.e. $P(t|c)^n$ for n occurrences of t, and the right edge corresponds to subsumption, i.e. $P(t|c)$ for any occurrence of t. The rows correspond to frequencies. The values $\frac{n}{1} \dots \frac{n}{n}$ in row n correspond to exponents, reflecting independence for $n/1 = n$ and subsumption for $n/n = 1$. The centre column (altitude) is half-way between independence and subsumption. Consequently, $n/(n+1)/2$ is half-way between independence and subsumption, and this leads to the probabilities for independent, semi-subsumed, and subsumed term occurrences:

Independent term occurrences	$P(t\|c)^n = P(t\|c)^{n/1}$
Semi-subsumed term occurrences	$P(t\|c)^{2n/(n+1)}$
Subsumed term occurrences	$P(t\|c)^1 = P(t\|c)^{n/n}$

The triangle in figure 2 and the table above underline how the notion of semi-subsumed events fits "neatly" into the traditional assumptions. The next section shows in a formal proof how the BM25 TF relates to the notion of semi-subsumed events.

4 BM25 TF: Subsequent Term Occurrences Are Semi-subsumed Events

The relationship between the BM25 TF and the notion of semi-subsumed events is not directly evident. Therefore, we prove now formally that the BM25 TF quantification assumes semi-subsumed term occurrences.

For an event occurring n times, $2n/(n+1)$ is the value in the altitude of the IST. This value is not equal to the BM25 TF quantification $\mathrm{tf}/(\mathrm{tf}+K)$. The common rewriting $(\mathrm{tf}/K)/(\mathrm{tf}/K+1)$ helps to establish the relationship between BM25 TF and semi-subsumed.

Theorem 1. *The BM25 TF quantification assumes the occurrences of a term to be semi-subsumed events, i.e. the subsequent occurrence of a term is more likely than if the occurrences were independent, and it is less likely than if they were subsumed.*

Proof. The probability for semi-subsumed events is $P(t|c)^{2n/(n+1)}$.

Set $n := \mathrm{tf}/K$, i.e. n is the normalised term frequency, where K is a normalisation factor (usually involving the pivoted document length). Then, the following equation holds:

$$P_D(t|c)^{2\cdot\mathrm{tf}/(\mathrm{tf}+K)} = P_D(t|c)^{2n/(n+1)} \tag{6}$$

The logarithmic form is $\sum_{t \in d \cap q} 2 \cdot \mathrm{tf}/(\mathrm{tf}+K) \cdot \mathrm{idf}(t,c)$. The constant 2 does not affect the ranking.

This proof finalises the contribution of this paper: The BM25 TF quantification assumes subsequent term occurrences (of the same term) to be semi-subsumed events.

5 Summary and Outlook

This paper introduced and discussed "semi-subsumed events". Semi-subsumed events overlap more than if the events were independent, and less than if they were subsumed. For the document-based, collection-wide term probability, $P_D(t|c)^n$ assumes *independence* of n occurrences of t, $P_D(t|c)^1$ assumes *subsumption*, and $P_D(t|c)^{2n/(n+1)}$ assumes *semi-subsumption*. The impact of semi-subsumed events is potentially beyond explaining the BM25 TF quantification. The wider impact is two-fold: on one hand the assumption semi-subsumed helps the theoreticians to develop probabilistic models with a precise semantics; on the other hand, making assumptions is essential for the pragmatic engineers to succeed in large-scale probabilistic reasoning. Regarding the semantics of probabilistic models, in many applications, there seems to be a "law of the series". The Dirichlet distribution and the Laplace law of succession address this law of the series, and future research is to relate Dirichlet and Laplace to semi-subsumed events. Also, the mid-point between disjoint and independent, i.e. semi-disjoint, is a special assumption and will be discussed in future work.

Acknowledgments. We would like to thank the reviewers for their helpful and inspiring comments and suggestions.

References

1. Robertson, S.: Understanding inverse document frequency: On theoretical arguments for idf. Journal of Documentation 60, 503–520 (2004)
2. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In: ACM SIGIR, pp. 232–241 (1994)
3. Robertson, S.E., Walker, S., Hancock-Beaulieu, M.: Large test collection experiments on an operational interactive system: Okapi at TREC. IP&M 31, 345–360 (1995)
4. Roelleke, T., Tsikrika, T., Kazai, G.: A general matrix framework for modelling information retrieval. IP&M, Special Issue on Theory in Information Retrieval 42(1) (2006)

Batch-Mode Computational Advertising Based on Modern Portfolio Theory

Dell Zhang and Jinsong Lu

Birkbeck, University of London
London WC1E 7HX, UK
dell.z@ieee.org, jingsong.lu@gmail.com

Abstract. The research on computational advertising so far has focused on finding the single best ad. However, in many real situations, more than one ad can be presented. Although it is possible to address this problem myopically by using a single-ad optimisation technique in serial-mode, i.e., one at a time, this approach can be ineffective and inefficient because it ignores the correlation between ads. In this paper, we make a leap forward to address the problem of finding the best ads in batch-mode, i.e., assembling the optimal set of ads to be presented altogether. The key idea is to achieve maximum revenue while controlling the level of risk by diversifying the set of ads. We show how the Modern Portfolio Theory can be applied to this problem to provide elegant solutions and deep insights.

1 Introduction

Online advertising has become a major industry. It is now an important source of income for many Web sites, particularly search engines such as Google and Yahoo!.

The research on computational advertising so far has focused on finding the *single* best ad [1]. However, in many real situations, more than one ad can be presented. For example, both Google and Yahoo! currently display up to 8 ads (sponsored links) for each query. Although it is possible to address this problem myopically by using a single-ad optimisation technique in *serial-mode*, i.e., one at a time, this approach can be ineffective and inefficient because it totally ignores the correlation among the "best" ads. While the selected ads all have high expected revenue, they can be very similar to each other, therefore displaying those ads is like "putting all eggs in one basket".

In this paper, we make a leap forward to address the problem of finding the best ads in *batch-mode*, i.e., assembling the optimal set of ads to be presented altogether. Our approach to batch-mode computational advertising is motivated by two observations: (1) the future revenues of ads are inherently uncertain; (2) the future revenues of ads are usually correlated with each other. The key idea is to achieve maximum revenue while controlling the level of risk by *diversifying* the set of ads. For example, given the query '`London weather`', even if the most profitable ads are all from companies selling umbrellas, it could be a better

L. Azzopardi et al. (Eds.): ICTIR 2009, LNCS 5766, pp. 380–383, 2009.

strategy for the search engine to show a mixture of ads from umbrella companies and sunscreen companies, because the revenue would be more stable. For another example, given the query '`fashion magazine`', men and women are probably looking for different products, therefore displaying some ads for men and some ads for women would give every user something relevant no matter what the gender is, and thus provide a better user experience overall and hopefully lead to an increase in revenue.

2 Approach

Assume that there are n ads $a_1, a_2, \ldots, a_n$ available in the advertising system. Given k ad places in the target Web page (either a search result page in 'sponsored search' or a content page in 'content matching'), the problem of batch-mode computational advertising is to select the optimal set of k ads.

We think this problem can be recast in the language of investment as follows. Each ad a_i is an *asset* (e.g., stock) with future *return* $r_i \in \mathbb{R}$ ($i = 1, \ldots, n$), which is determined by its bid price and click-through rate (CTR) following the popular pay-per-click (PPC) model. The CTR of each ad can be estimated from the historical data or approximated by the relevance of the ad to the query or the contextual page. The future return on a risky asset is inherently uncertain, so r_i should be regarded as a random variable. Suppose that the mean of r_i is $E(r_i) = \mu_i$ and the variance of r_i is $Var(r_i) = \sigma_i^2$. Moreover, let σ_{ij} be the covariance between r_i and r_j for all $1 \leq i, j \leq n$. It is well-known that $\sigma_{ii} = \sigma_i^2$, and when $i \neq j$ we have $\sigma_{ij} = \rho_{ij}\sigma_i\sigma_j$ where $\rho_{ij} \in [-1, 1]$ is the correlation coefficient between r_i and r_j. The covariance values can be estimated from the historical data or approximated using the pair-wise similarity of ads.

A set of k ads, S, can be considered as a *portfolio* of assets. Let a binary variable $b_i \in \{0, 1\}$ indicate whether a_i is selected: $b_i = 1$ if $a_i \in S$ or 0 otherwise. Let $w_i = b_i/k$, i.e., the fraction of the ad a_i in the portfolio. Then the overall future return of the portfolio, $r_p = \sum_{i=1}^n (b_i r_i)/k = \sum_{i=1}^n w_i r_i$, is characterised by its mean and variance: $E(r_p) = \sum_{i=1}^n w_i \mu_i$, $Var(r_p) = \sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij}$.

We would like to find the optimal portfolio that has the maximum return for a given risk, or equivalently the minimum risk for a desired return $E(r_p) = \mu_p$ of portfolio. Here the *risk* of portfolio is quantified by $Var(r_p)$: the less variance, the less volatility, the less risk.

$$\begin{aligned}
\text{minimize} \quad & \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n w_i w_j \sigma_{ij} \\
\text{subject to} \quad & \sum_{i=1}^n w_i \mu_i = \mu_p \\
& \sum_{i=1}^n w_i = 1 \\
& \forall 1 \leq i \leq n : w_i k \in \{0, 1\}
\end{aligned}$$

However, it turns out that the above combinatorial optimisation problem is NP complete and thus computational intractable. Therefore we relax the constraint to allow $w_i = b_i/k$ to take any real value in $\mathbb{R}$. The value of w_i can be considered as the weight of ad a_i. We first solve the following continuous optimisation problem to get the optimal weights, and then select the top k ads with highest weights as an approximation of the optimal portfolio.

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_i w_j \sigma_{ij} \\ \text{subject to} \quad & \sum_{i=1}^{n} w_i \mu_i = \mu_p \\ & \sum_{i=1}^{n} w_i = 1 \end{aligned}$$

In addition to making the computation feasible, we are now able to apply Modern Portfolio Theory (MPT) [2] to this problem to get elegant solutions and deep insights.

We can rewrite the above problem in matrix-vector form as follows:

$$\begin{aligned} \text{minimize} \quad & f(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{V}\mathbf{w} \\ \text{subject to} \quad & g_1(\mathbf{w}) = \mathbf{w}^T\mathbf{e} - \mu_p = 0 \\ & g_2(\mathbf{w}) = \mathbf{w}^T\mathbf{1} - 1 = 0 \ , \end{aligned}$$

where $\mathbf{w} = (w_1, w_2, \dots, w_n)^T$, $\mathbf{e} = (\mu_1, \mu_2, \dots, \mu_n)^T$, $\mathbf{1} = (1, 1, \dots, 1)^T$, and $\mathbf{V} \in \mathbb{R}^{n \times n}$ is the covariance matrix with $\mathbf{V}(i, j) = \sigma_{ij}$, $1 \le i, j \le n$. Using Lagrange multipliers, we can solve the above problem analytically to get the optimal vector of portfolio weights

$$\mathbf{w}_p = \frac{1}{D}(B\mathbf{V}^{-1}\mathbf{1} - A\mathbf{V}^{-1}\mathbf{e}) + \frac{1}{D}(C\mathbf{V}^{-1}\mathbf{e} - A\mathbf{V}^{-1}\mathbf{1})\mu_p \ ,$$

where $A = \mathbf{e}^T\mathbf{V}^{-1}\mathbf{1}$, $B = \mathbf{e}^T\mathbf{V}^{-1}\mathbf{e}$, $C = \mathbf{e}^T\mathbf{V}^{-1}\mathbf{1}$, and $D = BC - A^2$.

The above analytical solution is helpful in understanding the optimal portfolio, but it is computational expensive as it involves inversion of a dense matrix $\mathbf{V}$. In practice, we can use numerical computation techniques to get the numerical solution efficiently.

Every possible portfolio can be plotted in the risk-return space (with return μ_p on the y-axis and risk σ_p on the x-axis), and the collection of all such portfolios defines a region in this space. The hyperbola along the upper edge of this region is known as the *efficient frontier* (aka the Markowitz frontier), as illustrated in Fig 1. Combinations along this line represent portfolios for which there is lowest risk for a given level of return. Conversely, for a given amount of risk, the portfolio lying on the efficient frontier represents the combination offering the best possible return. The efficient frontier is the set of portfolios for which

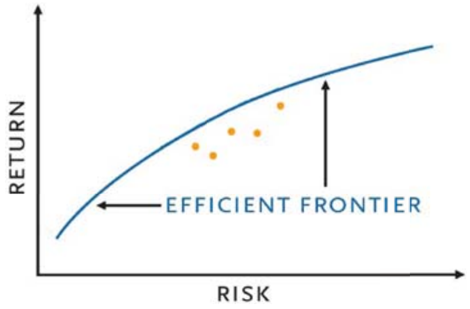


Fig. 1. The efficient frontier

one cannot improve both risk and return. On one hand, the region above the efficient frontier is unachievable by holding risky assets alone, i.e., no portfolios can be constructed corresponding to the points in this region. On the other hand, points below the frontier are suboptimal. Therefore a rational investor will hold a portfolio only on the frontier.

3 Conclusions

This paper presents a sketch theoretical development towards batch-mode computational advertising based on Modern Portfolio Theory (MPT). It is necessary to perform large scale experiments on real-world ad datasets to empirically evaluate our proposed approach, and compare it with existing heuristic methods in information retrieval for diversifying search results (such as MMR [3]). Furthermore, due to the sparsity of ad click-through data, how to estimate the future return of ads and their correlations effectively and efficiently remains to be an open research problem.

Acknowledgements

We would like to thank Dr Jun Wang (UCL) for interesting and fruitful discussions. Thanks also to the anonymous reviewers for their helpful comments.

References

1. Radlinski, F., Broder, A.Z., Ciccolo, P., Gabrilovich, E., Josifovski, V., Riedel, L.: Optimizing relevance and revenue in ad search: A query substitution approach. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Singapore, pp. 403–410 (2008)
2. Markowitz, H.M.: Portfolio selection. Journal of Finance 7(1), 77C–91C (1952)
3. Carbonell, J.G., Goldstein, J.: The use of MMR, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Melbourne, Australia, pp. 335–336 (1998)

Author Index