ADCS 2010

Proceedings of the Fifteenth Australasian Document Computing Symposium

10 December 2010

Edited by
Falk Scholer, Andrew Trotman and Andrew Turpin

Proceedings of the Fifteenth Australasian Document Computing Symposium

University of Melbourne, Melbourne, Victoria
10 December 2010

Published by
School of Computer Science and IT, RMIT University
Melbourne VIC 3001, Australia

Editors

Falk Scholer
Andrew Trotman
Andrew Turpin

ISBN: 978-1-921426-80-3
http://www.cs.rmit.edu.au/adcs2010

Proceedings of the Fifteenth Australasian Document Computing Symposium

University of Melbourne, Melbourne, Victoria
10 December 2010

Chairs' Preface

These proceedings contain the papers of the Fifteenth Australasian Document Computing Symposium hosted by the University of Melbourne.

The quality of submissions was again high this year. Of the 24 papers submitted, 10 were accepted for full presentation at the symposium (42%) and 5 were accepted as short papers (21%). The full written version of each submission received at least two anonymous reviews by independent, qualified international experts in the area; several received three reviews. Dual submissions were explicitly prohibited.

We would like to thank the members of the program committee and the extra reviewers for their efforts.

We would also like to thank Microsoft Research, NICTA (Victoria) and The University of Melbourne School of Engineering for their generous support of the event.

The symposium includes many formal presentations, but perhaps its greatest benefit lies in the opportunity it provides for document computing practitioners and researchers to get together and informally share ideas and enthusiasm. Once again we have collocated ADCS with The Australasian Language Technology Workshop 2010, sharing a joint paper session and social events.

Symposium Chair

Andrew Turpin	University of Melbourne

Programme Co-chairs

Falk Scholer	RMIT University
Andrew Trotman	University of Otago

Programme Committee

Vo Anh	University of Melbourne
Peter Bruza	Queensland University of Technology
Sally Jo Cunningham	University of Waikato
Shlomo Geva	Queensland University of Technology
David Hawking	Funnelback
Tim Jones	Australian National University
Judy Kay	University of Sydney
Yun Sing Koh	Auckland University of Technology
Alexander Krumpholz	CSIRO/ANU
Laurence Park	University of Western Sydney
Gitesh Raikundalia	Victoria University
Nathan Rountree	University of Otago
Tom Rowlands	CSIRO/ANU
Mark Sanderson	RMIT University
James A. Thom	RMIT University
Paul Thomas	CSIRO/ANU
William Webber	University of Melbourne
Mingfang Wu	RMIT University
Justin Zobel	University of Melbourne

Additional Reviewers

Bevan Koopman	Queensland University of Technology
Bhuva Lakshminarayanan	Queensland University of Technology
Richard O'Keefe	University of Otago
Raymond Wan	University of Kyoto

ADCS Steering Committee

Table of Contents

Front Matter

Full Papers

Short Papers

Extricating Meaning from Wikimedia Article Archives

Brian W. Curry, Andrew Trotman, and *Michael Albert*
Computer Science
University of Otago
Otago 9010 New Zealand
http://raiazome.com | (*andrew* | *malbert*)@*cs.otago.ac.nz*

Abstract *Wikimedia article archives (Wikipedia, Wiktionary, and so on) assemble open-access, authoritative corpora for semantic-informed datamining, machine learning, information retrieval, and natural language processing. In this paper, we show the `MediaWiki` wikitext grammar to be context-sensitive, thus precluding application of simple parsing techniques. We show there exists a worst-case bound on time complexity for all fully compliant parsers, and that this bound makes parsing intractable as well as constituting denial-of-service (DoS) and degradation-of-service (DegoS) attacks against all `MediaWiki` wikis. We show there exists a worse-case bound on storage complexity for fully compliant one-pass parsing, and that contrary to expectation such parsers are no more scalable than equivalent two-pass parsers. We claim these problems to be the product of deficiencies in the `MediaWiki` wikitext grammar and, as evidence, comparatively review 10 contemporary wikitext parsers for noncompliance with a partially compliant Parsing Expression Grammar (PEG).*

Keywords Document Standards, Information Retrieval, Web Documents, Wikipedia

1 Introduction

Wikimedia article archives assemble open-access, authoritative corpora for semantic-informed datamining, machine learning, information retrieval, and natural language processing. Unfortunately, these archives are described by an ad hoc document standard for which there exist no formal grammars for producing conformant parsers *or* conformant parsers beyond the de facto reference implementation, `Pywikipedia` [2].

In this paper, we show deficiencies in this standard to cause widespread non-compliance in third-party wikitext parsers, intractable storage and time complexity for all wikitext parsers (including `MediaWiki` itself) and viable denial-of-service (DoS) and degradation-of-service (DegoS) attacks against the most recent official release of `MediaWiki` as of this writing, `MediaWiki` 1.16.0.

Proceedings of the 16th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

As evidence of noncompliance, we comparatively review 10 contemporary wikitext parsers (including `MediaWiki` itself) against a partially compliant Parsing Expression Grammar (PEG) [6]. This review suggests there exists no fully compliant offline wikitext parser and only one fully compliant online wikitext parser *and it is not `MediaWiki` itself*: `Pywikipedia` [2]. Furthermore, the least compliantly parsed semantics are those we show induce worst-case intractability and insecurity in wikitext parsers.

As evidence of intractability, we present worst-case article archive input generally applicable to third party wikitext parsers. Analysis shows this input makes storage complexity prohibitive in disambiguation-compliant parsers and time complexity intractable in transclusion-compliant parsers.

As evidence of insecurity, we present worst-case article archive input specifically applicable to `MediaWiki` itself. Injecting this input into a local "clean-room" installation of the most recent official release of `MediaWiki` [1] shows this input makes all `MediaWiki` wikis susceptible to currently unresolved DoS and DegoS attacks, a compelling security flaw.

Finally, we present a partially compliant PEG. Constructed from exhaustive inspection of the `MediaWiki` codebase and real-world tests against local and remote `MediaWiki` wikis, this grammar matches and consumes most inter-article syntax (i.e., syntax conveying semantics *between* rather than *in* articles).

Due to its topical diversity, this paper's intended audience is threefold:

1. *Security consultants*, *system administrators* and *`MediaWiki`-invested policymakers*, given our exposure of prevailing vulnerabilities in the `MediaWiki` codebase (in Section 4).
2. *Information retrieval (IR)* and *natural language processing (NLP) specialists* as well as *`MediaWiki`-reliant dataminers*, given our findings of extensive noncompliance in official and third-party wikitext parsers (in Section 2) and appended publication of a partially compliant PEG (in the Appendix).
3. The *Wikimedia Foundation*, given our remedies of existing deficiencies in the Wikimedia article archive document standard (in Section 3).

	categorizations	disambiguations	occlusions	redirects	transclusions	wikilinks	interwikilinks
JWPL 0.453.4 [16]	*partial*	no	no	*partial*	*partial*	*partial*	*partial*
MediaWiki 1.16.0 [1]	**full**	no	**full**	**full**	**full**	**full**	**full**
mwlib (*7dbed545c6cd*) [11]	**full**	no	no	*partial*	**full**	*partial*	**full**
Parse::MediaWikiDump 1.0.6_01 [12]	*partial*	no	no	*partial*	no	no	no
Pywikipedia (*8616*) → *offline* [2]	*partial*	*partial*	**full**	*partial*	*partial*	*partial*	*partial*
Pywikipedia (*8616*) → *online* [2]	**full**	**full**	**full**	**full**	**full**	**full**	**full**
Yppy 0.0.8 [5]	**full**	*partial*	**full**	**full**	no	**full**	*partial*
Wiki2XML (*56074*) [9]	*partial*	no	*partial*	*partial*	*partial*	*partial*	*partial*
Wikipedia Miner (*92*) [10]	no	no	no	no	no	*partial*	no
Wikiprep 3.04 [7]	**full**	*partial*	*partial*	*partial*	*partial*	*partial*	*partial*
WWW:Wikipedia 1.97 [13]	no	no	no	no	no	*partial*	no

Table 1: Prevalence of fully compliant wikitext parsing

	fully parsed semantics	partially parsed semantics
Pywikipedia → *online*	**7**	*0*
MediaWiki	**6**	*0*
Yppy	**4**	*2*
mwlib	**3**	*2*
Pywikipedia → *offline*	**1**	*6*
Wikiprep	**1**	*6*
Wiki2XML	**0**	*6*
JWPL	**0**	*5*
Parse::MediaWikiDump	**0**	*2*
Wikipedia Miner	**0**	*1*
WWW:Wikipedia	**0**	*1*

Table 2: Parser compliance from Table 1

	fully compliant parsers	partially compliant parsers
disambiguations	**1**	*2*
occlusions	**3**	*2*
transclusions	**3**	*3*
interwikilinks	**3**	*4*
redirects	**3**	*5*
wikilinks	**3**	*6*
categorizations	**5**	*3*

Table 3: Semantic compliance from Table 1, ignoring offline Pywikipedia

2 Comparison

Comparative review of 10 contemporary wikitext parsers against our Appendix-presented grammar reveals common non-compliance. There exists no fully compliant offline parser and only one fully compliant online parser: Pywikipedia. Partially compliant parsers parsing most semantics include (in descending order of compliance): MediaWiki, Yppy and mwlib.

Table 1 summarizes this review. Rows signify reviewed parsers, columns reviewed semantics and row-column entries each parser's degree of compliance in parsing each semantic. Parser names are suffixed with the version or version control revision in parentheses we reviewed, preferring the latter for parsers whose most recent official release (as of this writing) was several months outdated or for which there was no official release (e.g., Pywikipedia).

We now discuss semantics, compliance issues associated with each and notable parsers compliantly addressing these issues.

2.1 Semantic compliance

Our review ignored all semantics other than those listed in the Table 1 header. Table 3 orders these semantics by ascending count of fully and partially compliantly parsed semantics in the first and second columns. Two of the three least compliantly parsed semantics produce worst-case bounds on wikitext parsing: *disambiguations* (producing prohibitive storage complexity for fully compliant one-pass parsing in Section 3.4) and *transclusions* (producing intractable time complexity for all fully compliant parsing in Sections 3.2 and 3.3). The remaining least compliantly parsed semantic, *occlusions*, is implicated in grammatical context-sensitivity (generating 75% of all context-sensitive productions in Section 3.1).

We first discuss *canonicalization,* a prerequisite for fully compliant parsing of 6 of our 7 reviewed semantics. Canonicalization reduces article titles in non-canonical to canonical form, enabling meaningful comparison between article titles regardless of form (e.g., a canonical article title ‘‘Talk:Hastur’’ refers to the same article as a non-canonical article title ‘‘TaLK__: hastur’’). There exist countably infinite non-canonical forms of each article title, so non-canonicalizing parsers return false negatives *and* positives by improperly matching non-canonical forms of the same article title as different articles. Canonicalizing parsers substitute, in article titles:

1. Embedded transclusions with their expansions.
2. Runs of space and underscore characters with a single space (e.g., "_ __" with " ").
3. Namespace aliases with the corresponding namespace name (e.g., "w:", "WP:", and "Project:" with "Wikipedia:").
4. Named-, decimal-, and hexadecimal-style HTML entities with the corresponding character (e.g., "&#230;" and "&aelig;" with "æ").
5. Subpage prefix "/" with the current article title within namespaces enabling the subpage feature (e.g., "/" with "Talk:Yuggoth" for all wikilinks in that article).
6. Autoformats with one or more canonical wikilinks to actual articles. This includes the *month day/month name* date autoformat, reformatted to read the opposite (e.g., [[15 March]] with [[March 15]]), and *ISO 8601* date autoformat, reformatted to read as two wikilinks (e.g., [[1890-08-20]] with "[[August 20]] [[1890]]").

We now discuss specific semantics.

Categorizations classify one article under another, denoted by double square brackets and language-specific Category namespace name (e.g., a string [[Category:Mythos]] classifies the current article under that category). Categorization compliance implies language-specific matching *and* canonicalization. Parsers disregarding categorizations confuse categorizations for wikilinks, since the two share the same notation.

Disambiguations are articles *transcluding* (see below) at least one language-specific disambiguation template (e.g., a string {{Disambig}} classifies the current article as a disambiguation). Disambiguation compliance implies preparsing the "MediaWiki:Disambiguationspage" article or equivalent metadata for the set of all disambiguation template names, matching those names *and* canonicalization. Parsers disregarding disambiguations assume redirects and wikilinks to disambiguations to be semantically meaningful, but they are not (e.g., a non-ambiguous wikilink [[Dagon]] to that article is semantically meaningful while an ambiguous wikilink [[Dagon (disambiguation)]] to that disambiguation is not).

Occlusions hide content from end users, denoted by XML 1.1-conformant **(a)** opening tag consisting of the '<' character, tag name, optional attributes and '>' character, **(b)** tag-specific text, and **(c)** closing tag consisting of the "</" string, same tag name and '>' character (e.g., "<pre>[[Shoggoth]]</pre>", which MediaWiki renders as the raw text [[Shoggoth]] rather than a wikilink to that article). Occlusion compliance implies matching this syntax *and* context-sensitively not matching any wikitext syntax in this syntax, as such tags *occlude* their content from conventional parsing. Parsers disregarding occlusions return false positives by improperly matching occluded wikitext.

Redirects symbolically link one article to another, denoted by #REDIRECT followed by a wikilink as the first wikitext for an article (e.g., a string "#REDIRECT [[Azathoth]]" redirects the current article to that). Redirect compliance implies matching *and* canonicalization. Parsers disregarding redirects confuse redirects for wikilinks, since the two share the same notation.

Transclusions dynamically expand one template's wikitext into the current article, denoted by double curly braces (e.g., a string {{Arkham}} expands that template's wikitext into the current article). Templates are articles under the Template namespace, so a transclusion {{Template_name}} actually transcludes an article named Template:Template_name. Transclusion compliance implies matching, canonicalization *and* fully recursive expansion. Parsers disregarding transclusions return false negatives by failing to expand transcluded wikitext. However, we show in Sections 3.2, 3.3 and 4 that the computational intractability of worst-case expansion makes such parsing inherently unsafe. Counter-intuitively, this implies that no transclusion compliance in a parser may be preferable to partial or full compliance.

Wikilinks link from one article to another on the same wiki, denoted by double square brackets (e.g., [[Yog-Sothoth]]). Wikilink compliance implies matching *and* canonicalization.

Interwikilinks link from one article to another on another wiki, denoted by double square brackets and a MediaWiki-recognized wiki name (e.g., [[craftywiki:Horror]]). Interwikilink compliance implies preparsing the "List of Wikipedias" and "Meta:Interwiki map" articles or equivalent metadata, matching *and* canonicalization – meriting distinction from mere wikilink compliance. Parsers disregarding interwikilinks confuse interwikilinks for wikilinks, since the two share the same notation. However, interwikilinks convey no meaningful semantics for most third-party parsers.

2.2 Parser compliance

Table 2 orders parsers by descending count of fully compliant (first) and partially compliant (second) semantics parsed. Three of the four most compliant parsers are PYTHON-implemented. All four of the least compliant parsers are JAVA- and PERL-implemented. We failed to find a working non-interpreted implementation, though *non*-working non-interpreted implementations do exist (e.g., FlexBisonParse [14]). This suggests multi-paradigmal, dynamically typed, interpreted languages to be ideal mediums for wikitext parsing.

Pywikipedia's full compliance is the collaborative result of its real-world use on MediaWiki-hosted wikis, the Wikimedia Toolserver and offline Wikimedia article archives. Our comparison is not entirely fair, therefore: Pywikipedia queries remote MediaWiki APIs for language-specific metadata retrieval, transclusion expansion and article validation. Denying Pywikipedia network access reduces its compliance to beneath that of mwlib – a more judicious comparison, perhaps.

Offline parsers have no access to comparable APIs, necessitating they statically populate data structures with a priori site-specific metadata *as well as* independently implement template preprocessors, article validators, etc. To accommodate this, we split Pywikipedia into “Pywikipedia → *offline*” (i.e., when offline) and “Pywikipedia → *online*” (i.e., when online).

MediaWiki’s lack of full compliance is a result of its inability to distinguish disambiguation from non-disambiguation articles, as has been noted at English Wikipedia itself [4].

3 Parser analysis

We expose deficiencies in the wikitext grammar and, for each deficiency, recommend amendments to existing Wikimedia policy.

3.1 Grammatical context-sensitivity

We now show the wikitext grammar to be context-sensitive, principally due to the presence of occlusions.

Suppose the wikitext grammar to be context-free. Then the production “*wikilink ← wikilink_begin wikilink_type wikilink_end*” context-freely matches wikilink [[R’lyeh]] in wikitext “<nowiki>[[R’lyeh]] </nowiki>”. However, MediaWiki parses wikilink syntax in nowiki tags as raw text rather than a wikilink. Then this production must match context-sensitively, a contradiction.

The nowiki tag occludes its wikitext content from conventional parsing. There exist 8 occluding tags: <!-...-!>, includeonly, nowiki, timeline, math, pre, source and syntaxhighlight. The latter 4 accept optional attributes; the remainder do not. Additionally, the closing tag corresponding to an opening occluding tag matches non-greedily (and hence context-sensitively).

This and the number of occluding tags complicate occlusion matching, as evidenced by the ratio of the number of occlusion productions to total number of productions κ. Of the 91 total productions, 30 involve occlusions. Of the 12 total context-sensitive productions, 9 involve occlusions. Then $\kappa \simeq 1/3$ in the set of all productions and $\kappa = 3/4$ in the set of context-sensitive productions, indicating occlusions dominate both. However, occlusions convey no meaningful semantics apart from their disabling of meaningful semantics!

As solution, we recommend Wikimedia distribute article archives encoding all *occluded* English punctuation as HTML entities (e.g., encoding “<nowiki>[[Cyaegha]]</nowiki>” as “<nowiki>&#91;&#91;Cyaegha&#93;&#93;</nowiki>”). This encoding is bijective and thus losslessly decodable on article archive deserialization. Since non-occlusion productions do not decode HTML entities, non-occlusion productions cannot match in HTML entity-encoded occlusion wikitext. Then given such an archive such productions are context-free. The resulting wikitext grammar may omit occlusion productions without concomitant loss of semantic compliance, and our PEG reduces to 61 total productions of which only 3 remain context-sensitive. It can be shown that these are also convertable to context-free productions, but only by breaking backward compatibility in the wikitext grammar.

3.2 Worst-case time complexity

We now show fully compliant parsers to suffer intractable worst-case time complexity $\mathrm{O}\left(|\mathfrak{M}|c^{|\mathfrak{T}|}\right)$, $|\mathfrak{M}|$ the number of non-template articles, $|\mathfrak{T}|$ the number of templates and c the maximum number of template transclusions per non-template article.

Consider the article archive consisting of at least two articles, all residing in the Main and Template namespaces and no others. The article set is partitionable into set $\mathfrak{M}$ on the Main non-templates and poset $\mathfrak{T}$ on the Template templates. Suppose non-templates have arbitrary titles and templates have minimal length titles, such that lexicographic comparison defines a well-ordering on $\mathfrak{T}$ (e.g., the first template is entitled ‘‘a’’, the second ‘‘b’’). Suppose non-template wikitext maximally transcludes the first template (i.e., maximally many repetitions of ‘‘{{a}}’’), template wikitext maximally transcludes the next template in $\mathfrak{T}$ for all templates except the last (e.g., template ‘‘a’’ wikitext is maximally many repetitions of ‘‘{{b}}’’), and the last template in $\mathfrak{T}$ terminally expands to binomially distributed single digit ‘0’ or ‘1’. Then all wikitext reduces to stochastic strings of ‘0’ and ‘1’ in the final expansion, and no transclusion expansion is losslessly memoizable. (Such expansions *are* memoizable if one does not mind the loss, as MediaWiki’s lossy memoization shows in Section 4.) Figure 1 depicts these assumptions with arrows signifying transclusion.

As wikitext length is bounded, the number of transclusions per article is bounded. Let c be this bound. Then each article comprises one node in the rooted tree of transclusions with fanout c where: **(a)** the first template in $\mathfrak{T}$ roots each such tree, **(b)** the last template in $\mathfrak{T}$ serves as the leaf nodes and **(c)** all other templates serve as internal nodes. All such trees are identical, so consider any tree r. By assumption r is complete of height $h = |\mathfrak{T}| - 1$. So r is size $s(c, |\mathfrak{T}|)$ given by

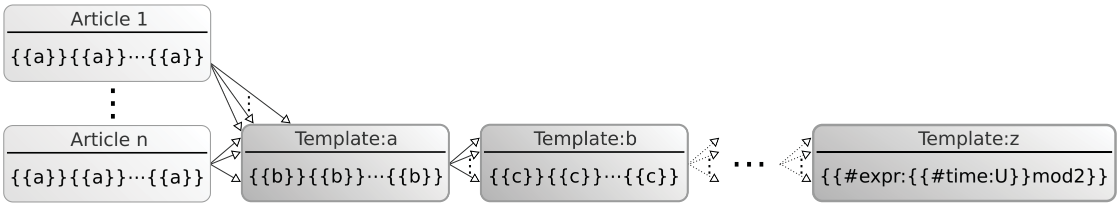


Figure 1: Article archive exhibiting worst-case time complexity

$$s(c, |\mathfrak{T}|) = \frac{c^{|\mathfrak{T}|} - 1}{c - 1} .$$

As each non-template recursively transcludes each such tree c times, the total number of transclusions $t(|\mathfrak{M}|, |\mathfrak{T}|)$ is given by

$$\begin{aligned} t(|\mathfrak{M}|, |\mathfrak{T}|) &= c|\mathfrak{M}|\, s(c, |\mathfrak{T}|) \\ &\in \mathrm{O}\left(\frac{c}{c-1}|\mathfrak{M}|\left(c^{|\mathfrak{T}|} - 1\right)\right) \\ &\in \mathrm{O}\left(|\mathfrak{M}| c^{|\mathfrak{T}|}\right) . \end{aligned}$$

As example, let $|\mathfrak{M}| = 6$ and $|\mathfrak{T}| = 26$ such that template titles iterate the alphabet for the worst-case article archive of 32 articles. Clearly, each template title consumes 1 byte and each transclusion 5 bytes. Backward compatibility requires wikitext length be practicably bounded to 32KB [3]. Then $c = \lfloor 32\text{KB}/5B \rfloor = 6553$, and

$$\begin{aligned} t(|\mathfrak{M}|, |\mathfrak{T}|) &\in \mathrm{O}\left(6 \cdot 6553^{26}\right) \\ &\in \mathrm{O}\left(10^{100}\right) . \end{aligned}$$

Parsing the worst-case article archive of only 32 articles expands on the order of a googol transclusions.

As solution, we recommend precaution in the `MediaWiki` codebase against pathological template abuse. Given the improbability of third-party reimplementations of fully compliant *and* precautionary transclusion preprocessors, we recommend Wikimedia distribute one additional article archive for each site

1. comprising all articles except those residing in the `Template` namespace, and
2. expanding transclusions in all article wikitext “in place” at archive creation time.

3.3 Worst-case time complexity (revisited)

We established a template-dependent worst-case time complexity for fully compliant wikitext parsing in the previous Section. Now we revisit the issue with a worst-case grammar establishing a grammar-dependent worst-case time complexity and showing the latter to aggravate constant costs associated with the former, producing an aggregate worst-case time complexity.

As the Appendix discusses, disambiguation-compliant grammars are archive-specific. They remain incomplete until parsing the ‘‘`MediaWiki:Disambiguationspage`’’ article, after which the DISAMBIGUATION production may be specified to match all archive-specific disambiguation template names and thereby complete the grammar. The size of this production and thus this grammar is a function of the number of such names.

As wikitext length is bounded, the number of disambiguation template names referenced in ‘‘`MediaWiki:Disambiguationspage`’’ wikitext is bounded. Let d be this bound. Then identifying disambiguation articles requires eligible transclusions (i.e., transclusions in wikitext residing in the `Main` namespace) be iteratively tested against d alternatives.

Recall that each non-template in the worst-case article archive of the previous Section consisted of maximally many such transclusions. Append a ‘‘`MediaWiki:Disambiguationspage`’’ article referencing maximally many disambiguation template names to this archive. Then the total cost of parsing transclusions $T(|\mathfrak{M}|, |\mathfrak{T}|)$ is given by

$$\begin{aligned} T(|\mathfrak{M}|, |\mathfrak{T}|) &= d \cdot t(|\mathfrak{M}|, |\mathfrak{T}|) \\ &\in \mathrm{O}\left(\frac{dc}{c-1}|\mathfrak{M}|\left(c^{|\mathfrak{T}|} - 1\right)\right) \\ &\in \mathrm{O}\left(t(|\mathfrak{M}|, |\mathfrak{T}|)\right) . \end{aligned}$$

Parsing this article archive expands the same order of transclusions as the prior, but amplifies the constant cost associated with doing so. We now show these constants to be non-negligible.

As stated wikitext length is bounded to 32KB. Suppose disambiguation template names are 2 bytes in length. In ‘‘`MediaWiki:Disambiguationspage`’’ wikitext, the itemization of each such name requires a 12 byte prefix ‘‘`*[[Template:`’’ and 3 byte suffix ‘‘`]]\n`’’ for automated bot discovery. Then each such item consumes 17 bytes and $d = \lfloor 32\text{KB}/17\text{B} \rfloor = 1927$, certainly within the range of 2 byte template names. Incorporating non-negligible constants, aggregate worst-case time complexity $T(|\mathfrak{M}|, |\mathfrak{T}|)$, $|\mathfrak{M}|$ the number of `Main` namespace articles and $|\mathfrak{T}|$ the number of `Template` namespace articles, is

$$T(|\mathfrak{M}|, |\mathfrak{T}|) = 1927 \cdot \mathrm{O}\left(|\mathfrak{M}| 6553^{|\mathfrak{T}|}\right) .$$

As solution, we recommend Wikimedia eliminate article-specific circular dependencies in the wikitext grammar. To do so for disambiguations, we propose the `#DISAMBIG` pragma explicitly declaring an article to be a disambiguation page. This pragma maintains backward compatibility with `MediaWiki` syntax, third-party parsers and the article corpus itself by requiring this “magic word” prefix be suffixed with a disambiguation transclusion (e.g., by supplanting all instances of “`{{Disambig}}`” in English Wikipedia with “`#DISAMBIG {{Disambig}}`”). This pragma also adds explicit invariance to the existing wikitext grammar: namely, that each disambiguation page be associate with one and only one disambiguation template. In the existing wikitext grammar, disambiguation pages may be associate with no such template (by explicitly categorizing themselves under `[[Category:Disambiguation]]` rather than transcluding such a template) or more than one (by transcluding more than one, in which case the resulting disambiguation is inconsistent). This has the beneficial by-product of eliminating additional context-sensitivity from the wikitext grammar, which when coupled with the recommendation of Section 3.1 reduces the number of context-sensitive productions to 2. For alternative solution, see English Wikipedia's “`Wikipedia: Disambiguation pages aren't articles`”.

3.4 Worst-case one-pass complexity

We now show fully compliant one-pass parsers to suffer prohibitive worst-case storage $\mathrm{O}(|\mathfrak{N}|)$, $|\mathfrak{N}|$ the number of articles, and scale no better than equivalent two-parse parsers in this case. *To exhibit these inefficiencies, this Section presents worst-case article archive input orthogonal to that of Section 3.3. While the two could be profitably composited into another aggregate worst case, that does not substantially revise the conclusion of this Section.*

Consider the article archive consisting of poset $\mathfrak{N} = (\mathfrak{A}_1, \mathfrak{A}_2, \ldots, \mathfrak{A}_{|\mathfrak{A}|}, \mathfrak{G}_1, \mathfrak{G}_2, \ldots, \mathfrak{G}_{|\mathfrak{G}|})$, $\mathfrak{G}$ the non-empty set of grammar-generative articles comprising at least “`MediaWiki:Disambiguationspage`”, “`List of Wikipedias`” and “`Meta:Interwiki map`” and $\mathfrak{A}$ the non-empty set of all remaining articles. Since the number of grammar-generative articles (3 under our PEG) is substantially smaller than the number of non-grammar-generative articles (3,447,220 for English Wikipedia as of this writing), $|\mathfrak{G}| \ll |\mathfrak{A}| \simeq |\mathfrak{N}|$.

Suppose all wikitext in $\mathfrak{A}$ consists of maximally many transclusions and/or wikilinks containing a colon. Then each such transclusion ambiguously signifies a possible disambiguation and each such wikilink a possible interwikilink or interlanguagelink. Since no parser may certify which is which until having parsed all wikitext in $\mathfrak{G}$, the resulting article archive exhibits *maximal semantic ambiguity*.

As example, the wikilink `[[Yog: Sothoth]]` ambiguously signifies either **(a)** a wikilink to that article, **(b)** an interwikilink to article “`Sothoth`” on the external wiki identified by interwiki prefix “`yog`” or **(c)** an interlanguagelink to the same article on the external Wikimedia wiki identified by language code “`yog`” (e.g., `http://yog.wikipedia.org/wiki/Sothoth`).

Suppose we implement a two-pass parser naïvely resolving these ambiguities as follows:

1. In the first pass, linearly search the article archive for all articles in $\mathfrak{G}$ ignoring all wikitext except that in $\mathfrak{G}$. These are the last articles in $|\mathfrak{N}|$, incurring storage cost $\mathrm{O}(|\mathfrak{G}|)$ and time cost $\mathrm{O}(|\mathfrak{N}|)$. Then generate the archive-specific grammar required for fully compliant parsing.
2. In the second pass, linearly parse all wikitext given this grammar, incurring time cost $\mathrm{O}(|\mathfrak{N}|)$.

Then the naïve two-pass parser suffers worst-case storage $\mathrm{O}(|\mathfrak{G}|)$ and time $2\mathrm{O}(|\mathfrak{N}|) = \mathrm{O}(|\mathfrak{N}|)$. Suppose we optimize this into a one-pass parser as follows:

1. Linearly parse wikitext until parsing all wikitext in $\mathfrak{G}$, caching all semantically ambiguous wikitext for subsequent reparsing. Since all wikitext in $\mathfrak{A}$ exhibits maximal semantic ambiguity, this incurs storage *and* time cost $\mathrm{O}(|\mathfrak{N}| - |\mathfrak{G}|)$.
2. Parse all wikitext in $\mathfrak{G}$ to generate the archive-specific grammar, incurring storage *and* time cost $\mathrm{O}(|\mathfrak{G}|)$.
3. Reparse all cached wikitext given this grammar, incurring time cost $\mathrm{O}(|\mathfrak{N}| - |\mathfrak{G}|)$.

Then the optimized one-pass parser suffers worst-case storage $\mathrm{O}(|\mathfrak{N}| - |\mathfrak{G}|) + \mathrm{O}(|\mathfrak{G}|) = \mathrm{O}(|\mathfrak{N}|)$, substantially worse than that of naïve two-pass parsing, and time $\mathrm{O}(|\mathfrak{N}| - |\mathfrak{G}|) + \mathrm{O}(|\mathfrak{G}|) + \mathrm{O}(|\mathfrak{N}| - |\mathfrak{G}|) \simeq \mathrm{O}(|\mathfrak{N}|)$, equivalent to that of naïve two-pass parsing.

Both parsers assume no prior indexing of compressed article archive input. We note in passing that pseudo-indexing is technically feasible: present-day Wikimedia article archives are bzip2-compressed files internally partitioned into blocks of default size 900KB, which while disallowing random access to exact byte offsets do allow random access to exact block offsets [8]. Indexing article title to 2-element tuple (b, y), b the offset to the compressed block in which that article begins and y the offset to that article's first uncompressed byte in that block, during one-pass parsing *could* reduce the real-world storage cost (by avoiding caching) at some additional time cost (by forcing re-decompressed seeking of on-disk blocks). Further research required.

Consider English Wikipedia, whose 12 GB archive `enwiki-20101011-pages-meta-current.xml.bz2` uncompresses to approximately $160 - 230$ GB. Then worst-case storage $\mathrm{O}(|\mathfrak{N}|)$ is prohibitive on high-volume archives and there exist compelling incentives not to implement one-pass parsers *without also implementing pseudo-indexing*.

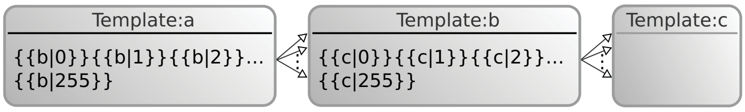


Figure 2: Article archive exhibiting our Denial-of-Service (DoS) exploit

> As solution, see our solution to the problem of Section 3.3.

4 Worst-case exploitation

Denial-of-service (DoS) attacks render computing services unavailable to end users by exploiting hardware-, protocol- and application-level insecurities. Degradation-of-service (DegoS) attacks render such services non-performant by brute-force consumption of scarce computing resources (e.g., bandwidth, CPU load). As a practical demonstration, we now improve the worst-case article archive input of Section 3.2 into viable DoS and DegoS attacks on `MediaWiki` itself.

We tested these attacks against clean-room installation of the most recent official releases of `MediaWiki` (1.16.0), `MySQL` (5.1.50), `PHP` (5.3.3), the `Apache` HTTP Server (2.2.16) and `Linux` (2.6.36) as of this writing, where "clean-room" means:

1. No optional `MediaWiki` extensions and only those `PHP` extensions required by `MediaWiki`.
2. Default `MediaWiki`, `MySQL` and `PHP` settings.
3. Stock `Apache` HTTP Server and `Linux` kernel modules and settings.

We expect these attacks retroactively apply to *all* `MediaWiki` wikis regardless of version, configuration or system. However, we verified this neither locally or remotely against publicly accessible wikis. (In the latter case, doing so violates ISP and University acceptable use policies as well as constituting imprisonable offenses under domestic and international law).

4.1 DoS attack

While worst-case article archive input of Section 3.2 makes third-party parsing of article archives intractable, its memoization by the `MediaWiki` preprocessor makes this input inadequate for attacks on `MediaWiki` itself.

`MediaWiki` memoizes *all transclusions of the same template and same template parameters in the same article* to the first expansion of the transclusion, even with transclusions expanding differently! As example, a `Template:Yog` with wikitext `{{CURRENTTIME}}` *should* expand differently in an article with wikitext ''`{{Yog}}{{Yog}}{{Yog}}`'' when the current time in seconds changes between the first and second or second and third expansion of that template. Of course, this is not what happens; `MediaWiki` forcefully sets the second and third expansion to the first regardless.

`MediaWiki` memoization negates the usefulness of our prior worst-case input. However by the above invariant, `MediaWiki` memoizes no transclusions of the same template *and different template parameters* in the same article. Then altering this input so the last template in $|\mathfrak{T}|$ is empty and all transclusions in all other templates in $|\mathfrak{T}|$ are uniquely parametrized within their templates prevents memoization.

`PHP` prematurely terminates scripts exceeding its *max_execution_time*, defaulting to 30s. So there exists some least total number of transclusions t_1 for which `MediaWiki` exceeds this setting. Testing shows $t_1 \in \left(2^{15}, 2^{16}\right]$ for our local installation, so assume $t_1 = 2^{16}$ for convenience. But $256^2 = 2^{16}$, so 2 templates of 256 transclusions each induces `PHP` to prematurely terminate `MediaWiki`. Figure 2 depicts these assumptions with arrows signifying transclusion.

Submitting templates ''`b`'' and ''`c`'' to the target `MediaWiki` wiki does not trigger the attack; submitting template ''`a`'' does. Then the attack consists of first submitting the former two templates once each, then repetitively resubmitting the latter template. Each resubmission starves the target `MediaWiki` wiki with near 100% CPU load for exactly 30s, after which `PHP` interrupts the submission, `MySQL` rolls back the transaction and `MediaWiki` resumes receiving queries at normal CPU load. Then the attacker resubmits template ''`a`'' and the attack resumes.

Each resubmission enjoys a payload of only 2.2KB. So the attack is inherently asymmetric: a milliseconds worth of effort on the attacking machine generates 30s of high CPU load on the attacked machine. But 16 articles of 2 transclusions each also induces `MediaWiki` to exceed `PHP`'s *max_execution_time* setting, so the payload is reducible to 15B. The simplicity of this asymmetry lends itself to anonymous distribution via decentralized botnets [15], thus extending its scope to (largely) non-targetable resilient attack networks.

4.2 DegoS attack

However, a subversive alternative suggests itself: induce the target `MediaWiki` wiki to spin-wait *itself*.

There exists some greatest total number of transclusions t_0 for which `MediaWiki` does not exceed `PHP`'s *max_execution_time*. Then $t_0 = t_1 - 1$ when t_1 is known or the greatest number t_1 is known to be strictly greater than when t_1 is not known. Larger values induce longer downtime, so the former is preferable. $t_1 > 2^{15}$ in our case, so let $t_0 = 2^{14} = 128^2$.

We measured 2 templates of 128 transclusions each to consume $28-30$s of wall clock time per submission of template ``a'', allowing MySQL to successfully complete submission transactions. Then submitting template ``a'' of 128 transactions of ``b'', template ``b'' of 128 transactions of ``c'' and template ``c'' empty as before initiates this attack. The attacker identifies high-edit articles, then injects one malicious transclusion of template ``a'' into each article – preferably adjacent to or embedded in existing transclusions in each article *and/or* accompanied by one or more seemingly benign edits to each article as a cloaking measure. Since each transclusion expands to nothing, search engines show no evidence of the attack. Since each transclusion consists of only 5B in high-edit articles consisting of up to 32KB, each article shows little evidence of the attack. Since each transclusion expansion in each article consumes the maximal amount of wall clock time without triggering MediaWiki, PHP, Apache, or kernel defenses, all subsequent edits on each article suffer the same spin-wait, and the DegoS attack is described.

> As solution, we recommend Wikimedia implement safeguards against transclusion fanout in the MediaWiki codebase *and that all third-party parsers immediately follow suite*.

5 Conclusion

Parsing Wikimedia article archives has been shown to be non-trivial. Worst-case article archive input of maximal recursive transclusion renders parsing computationally intractable. Worst-case article archive input of maximal semantic ambiguity renders one-pass parsing storage prohibitive. Average-case input of context-sensitive occlusions, non-expanded transclusions, non-canonical wikilinks and ambiguous disambiguations and interwikilinks complicates parser compliance irrespective of worst case complexity.

Susceptibility of MediaWiki wikis to transclusion-enabled DoS and DegoS attacks suggests transclusion-ignoring parsers to be fundamentally more secure than transclusion-compliant parsers. For safety, third-party parsers attempting to recursively expand tranclusions must duplicate existing provisions against transclusion abuse in the MediaWiki codebase *as well as devise new provisions against these novel attacks*.

Comparative reviewal of 10 contemporary wikitext parsers reveals widespread syntactic and semantic non-compliance. As expected, the least compliantly parsed semantics match those responsible for aforementioned worst-case bounds. We recommend a parser-focused redress of Wikimedia article archive policies and of the MediaWiki wikitext grammar, as follows: **(a)** encode in-occlusion punctuation as HTML entities; **(b)** declare disambiguations via #DISAMBIG pragmas; **(c)** recursively expand transclusions in-place; **(d)** publicize safeguards against transclusion abuse.

References

[1] Various authors. MediaWiki 1.16.0. http://www.mediawiki.org, July 2010.

[2] Various authors. Pywikipedia (svn revision 8616). http://pywikipediabot.sourceforge.net, October 2010.

[3] Various authors. Wikipedia article size. http://en.wikipedia.org/wiki/Wikipedia:Article_size, September 2010.

[4] Various authors. Wikipedia disambiguation pages aren't articles. http://en.wikipedia.org/wiki/Wikipedia:Disambiguation_pages_aren't_articles, May 2010.

[5] Brian W. Curry. Yppy 0.0.8. http://bitbucket.org/leycec/yppy, October 2010.

[6] Bryan Ford. Parsing expression grammars: a recognition-based syntactic foundation. *ACM SIGPLAN Notices*, Volume 39, Number 1, pages 111–122, 2004.

[7] Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of The 20th International Joint Conference on Artificial Intelligence*, Hyderabad, India, January 2007.

[8] Interiot. Random access [*on*] bzip2[*-compressed article archives*]. http://meta.wikimedia.org/wiki/User:Interiot/Random_access_bzip2, May 2007.

[9] Magnus Manske. Wiki2XML (svn revision 56074). http://toolserver.org/~magnus/wiki2xml/w2x.php, September 2009.

[10] David Milne. An open-source toolkit for mining Wikipedia. In *New Zealand Computer Science Research Student Conference (NZCSRSC) 2009 Proceedings*, Auckland, New Zealand, April 2009.

[11] PediaPress. mwlib (git revision 7dbed545c6cd). http://code.pediapress.com/wiki/wiki/mwlib, October 2010.

[12] Tyler Riddle. Parse::MediaWikiDump 1.0.6_01. http://search.cpan.org/dist/Parse-MediaWikiDump, June 2010.

[13] Ed Summers and Brian Cassidy. WWW::Wikipedia 1.97. http://search.cpan.org/dist/WWW-Wikipedia/, June 2010.

[14] Timwi and Magnus Manske. FlexBisonParse (svn revision 71620). http://svn.wikimedia.org/viewvc/mediawiki/trunk/parsers/flexbisonparse, August 2010.

[15] Ryan Vogt, John Aycock and Michael J. Jacobson, Jr. Army of botnets. In *Network and Distributed System Security Symposium*. Internet Society, 2007.

[16] Torsten Zesch, Christof Müller and Iryna Gurevych. Extracting lexical semantic knowledge from Wikipedia and Wiktionary. In *Proceedings of the Conference on Language Resources and Evaluation*, Marrakech, Morocco, May 2008.

Appendix

This Appendix presents a partially compliant wikitext grammar for programmatically generating article archive parsers. We showed this grammar to be context-sensitive in Section 3.1, so context-free metalanguages such as Backus-Naur Form (BNF) do not apply. Instead, we apply the recently developed Parsing Expression Grammar (PEG) metalanguage to context-sensitively describe this grammar [6].

Page constraints, readability concerns and initial application to the production of wikilink graphs make this PEG only partially compliant. It parses most *inter*-article semantics (relating two or more articles) but no *intra*-article semantics (relating the structure within an article). This includes all *categorization*, *disambiguation*, *redirect*, *wikilink*, and *interwikilink* semantics as well as some *transclusion* semantics, and none else.

Productions in **boldface** are language-specific. They remain unset until after parsing revelant metadata relevant from the archive preamble. When the article archive preamble fails to provide such metadata (e.g., for the ***wikilink_day_month_month*** production), a parser either requires a priori knowledge of the language under inspection or must delete all such productions and productions requiring these productions (e.g., the *wikilink_day_month* production). By default, language-specific productions in the grammar below assume English Wikipedia.

Productions in SMALL CAPS are archive-specific. They remain unset until after parsing relevant articles from the archive body. Such articles are *grammar-generative*, in that their wikitext assists the parser to generate itself. Prior to parsing the set of all grammar-generative articles, the grammar is incompletely generated. In this incomplete state, wikitext potentially matching one or more unset productions cannot be reliably consumed and must either be discarded or cached for subsequent reparsing. Parsers performing the former are necessarily two-pass; parsers performing the latter are one-pass. In either case, compliant parsers must iteratively bootstrap themselves in a language-specific manner to eventual completion.

Productions split by "←" are directly context-free. Productions split by "↩" are directly context-sensitive. Conveniently, most productions are the former.

We invite the interested reader to review `Yppy`[5], open-source graph theoretic software implementing this formalism as Python-compatible regular expressions.

```
                 wikitext  ↩  redirect? (occlusion | transclusion | wikilink | .)
                 redirect  ←  redirect_prefix redirect_begin wikilink wikilink_end
          redirect_prefix  ←  whitespace* redirect_magic_word whitespace* ":"? whitespace*
      redirect_magic_word  ←  "#" [Rr] [Ee] [Dd] [Ii] [Rr] [Ee] [Cc] [Tt]
           redirect_begin  ↩  wikilink_begin | . redirect_begin
                occlusion  ←  includeonly_tag | nowiki_tag | timeline_tag | comment_tag |
                              math_tag | pre_tag | source_tag | syntaxhighlight_tag
          includeonly_tag  ←  includeonly_tag_open includeonly_tag_body
     includeonly_tag_open  ←  tag_begin "includeonly" tag_end
     includeonly_tag_body  ↩  tag_begin "/includeonly" tag_end | . includeonly_tag_body
               nowiki_tag  ←  nowiki_tag_open nowiki_tag_body
          nowiki_tag_open  ←  tag_begin "nowiki" tag_end
          nowiki_tag_body  ↩  tag_begin "/nowiki" tag_end | . nowiki_tag_body
             timeline_tag  ←  timeline_tag_open timeline_tag_body
        timeline_tag_open  ←  tag_begin "timeline" tag_end
        timeline_tag_body  ↩  tag_begin "/timeline" tag_end | . timeline_tag_body
              comment_tag  ←  comment_tag_open comment_tag_body
         comment_tag_open  ←  html_entity_less_than "!-"
         comment_tag_body  ↩  "-" html_entity_greater_than | . comment_tag_body
                 math_tag  ←  math_tag_open math_tag_body
            math_tag_open  ←  tag_begin "math" tag_end_attributes
            math_tag_body  ↩  tag_begin "/math" tag_end | . math_tag_body
                  pre_tag  ←  pre_tag_open pre_tag_body
             pre_tag_open  ←  tag_begin "pre" tag_end_attributes
             pre_tag_body  ↩  tag_begin "/pre" tag_end | . pre_tag_body
               source_tag  ←  source_tag_open source_tag_body
          source_tag_open  ←  tag_begin "source" tag_end_attributes
          source_tag_body  ↩  tag_begin "/source" tag_end | . source_tag_body
      syntaxhighlight_tag  ←  syntaxhighlight_tag_open syntaxhighlight_tag_body
 syntaxhighlight_tag_open  ←  tag_begin "syntaxhighlight" tag_end_attributes
 syntaxhighlight_tag_body  ↩  tag_begin "/syntaxhighlight" tag_end | . syntaxhighlight_tag_body
                tag_begin  ←  html_entity_less_than
                  tag_end  ←  whitespace* html_entity_greater_than
       tag_end_attributes  ←  tag_end_attribute* whitespace* html_entity_greater_than
        tag_end_attribute  ←  whitespace+
                              [a-zA-Z] ([a-zA-Z0-9] | "-" | "_" | ".")* '="' (!'"' .)* '"'
             transclusion  ←  transclusion_begin transclusion_body transclusion_end
       transclusion_begin  ←  "{{" wikilink_whitespace* transclusion_begin_subst?
 transclusion_begin_subst  ←  [Ss] [Uu] [Bb] [Ss] [Tt] ":"
         transclusion_end  ←  wikilink_whitespace* "}}"
        transclusion_body  ↩  disambiguation | transclusion | . transclusion_body
           DISAMBIGUATION  ←  Unset prior to parsing the ‘‘MediaWiki:Disambiguationspage’’ article.
                 wikilink  ←  wikilink_begin wikilink_type wikilink_end
           wikilink_begin  ←  wikilink_open wikilink_whitespace*
             wikilink_end  ←  wikilink_whitespace* wikilink_close
            wikilink_open  ←  "[["
           wikilink_close  ←  "]]"
      wikilink_whitespace  ←  " " | "_"
            wikilink_type  ↩  wikilink_category | wikilink_day_month | wikilink_iso_8601 | wikilink_normal
        wikilink_category  ←  wikilink_namespace_category wikilink_qualifier_end wikilink_body
       wikilink_day_month  ←  wikilink_day_month_day wikilink_whitespace* wikilink_day_month_month
   wikilink_day_month_day  ←  [0-9] [0-9]?
 wikilink_day_month_month  ←  [Jj] "anuary" | [Ff] "ebruary" | [Mm] "arch" | [Aa] "pril" |
                              [Mm] "ay" | [Jj] "une" | [Jj] "uly" | [Aa] "ugust" |
                              [Ss] "eptember" | [Oo] "ctober" | [Nn] "ovember" |
                              [Dd] "ecember"
        wikilink_iso_8601  ←  wikilink_iso_8601_year "-"
                              wikilink_iso_8601_month "-" wikilink_iso_8601_day
   wikilink_iso_8601_year  ←  [0-9] [0-9] [0-9] [0-9]
```

wikilink_iso_8601_month ← `[0-9] [0-9]`

wikilink_iso_8601_day ← `[0-9] [0-9]`

wikilink_normal ← (wikilink_subpage_prefix | wikilink_qualifier)? wikilink_body

wikilink_subpage_prefix ← `"/"` wikilink_whitespace*

wikilink_qualifier ← wikilink_qualifier_prefix? wikilink_qualifier_body wikilink_qualifier_end

wikilink_qualifier_prefix ← `":"` wikilink_whitespace*

wikilink_qualifier_body ← wikilink_namespace | wikilink_interlanguage | wikilink_interwiki

wikilink_qualifier_end ← wikilink_whitespace* `":"` wikilink_whitespace*

wikilink_namespace ← wikilink_namespace_main | wikilink_namespace_talk |
wikilink_namespace_user | wikilink_namespace_user_talk |
wikilink_namespace_wikipedia | wikilink_namespace_wikipedia_talk |
wikilink_namespace_file | wikilink_namespace_file_talk |
wikilink_namespace_mediawiki | wikilink_namespace_mediawiki_talk |
wikilink_namespace_template | wikilink_namespace_template_talk |
wikilink_namespace_help | wikilink_namespace_help_talk |
wikilink_namespace_category | wikilink_namespace_category_talk |
wikilink_namespace_special | wikilink_namespace_media

wikilink_namespace_main ← `[Mm] [Aa] [Ii] [Nn]`

wikilink_namespace_talk ← `[Tt] [Aa] [Ll] [Kk]`

wikilink_namespace_user ← `[Uu] [Ss] [Ee] [Rr]`

wikipedia_namespace_user_talk ← wikilink_namespace_user wikilink_namespace_talk_end

wikilink_namespace_wikipedia ← `[Ww] [Ii] [Kk] [Ii] [Pp] [Ee] [Dd] [Ii] [Aa]` |
`[Pp] [Rr] [Oo] [Jj] [Ee] [Cc] [Tt]` |
`[Ww] [Pp]` | `[Ww]`

wikilink_namespace_wikipedia_talk ← (`[Ww] [Ii] [Kk] [Ii] [Pp] [Ee] [Dd] [Ii] [Aa]` |
`[Pp] [Rr] [Oo] [Jj] [Ee] [Cc] [Tt]`) wikilink_namespace_talk_end |
`[Ww] [Tt]`

wikilink_namespace_file ← `[Ff] [Ii] [Ll] [Ee] | [Ii] [Mm] [Aa] [Gg] [Ee]`

wikilink_namespace_file_talk ← wikilink_namespace_file wikilink_namespace_talk_end

wikilink_namespace_mediawiki ← `[Mm] [Ee] [Dd] [Ii] [Aa] [Ww] [Ii] [Kk] [Ii]`

wikilink_namespace_mediawiki_talk ← wikilink_namespace_mediawiki wikilink_namespace_talk_end

wikilink_namespace_template ← `[Tt] [Ee] [Mm] [Pp] [Ll] [Aa] [Tt] [Ee]`

wikilink_namespace_template_talk ← wikilink_namespace_template wikilink_namespace_talk_end

wikilink_namespace_help ← `[Hh] [Ee] [Ll] [Pp]`

wikilink_namespace_help_talk ← wikilink_namespace_help wikilink_namespace_talk_end

wikilink_namespace_category ← `[Cc] [Aa] [Tt] [Ee] [Gg] [Oo] [Rr] [Yy]`

wikilink_namespace_category_talk ← wikilink_namespace_category wikilink_namespace_talk_end

wikilink_namespace_special ← `[Ss] [Pp] [Ee] [Cc] [Ii] [Aa] [Ll]`

wikilink_namespace_media ← `[Mm] [Ee] [Dd] [Ii] [Aa]`

wikilink_namespace_talk_end ← wikilink_whitespace+ wikilink_namespace_talk

WIKILINK_INTERLANGUAGE ← Unset prior to parsing the ‘‘`List of Wikipedias`’’ article.

WIKILINK_INTERWIKI ← Unset prior to parsing the ‘‘`Meta:Interwiki map`’’ article.

wikilink_body ← (!wikilink_invalid_char .)+ wikilink_anchor? wikilink_label?

wikilink_anchor ← wikilink_whitespace* `"#"` (!wikilink_invalid_char .)*

wikilink_label ← wikilink_whitespace* `"|"` (!wikilink_label_invalid_char .)*

wikilink_invalid_char ← `"#"` | `"<"` | `">"` | `"["` | `"]"` | `"|"` | `"{"` | `"}"` | `"\t"` | `"\n"`

wikilink_label_invalid_char ← wikilink_close | `"\t"` | `"\n"`

html_entity_less_than ← `"&lt;"`

html_entity_greater_than ← `"&gt;"`

whitespace ← `" " | "\t" | "\n"`

Seeing the forest from trees : Blog Retrieval by Aggregating Post Similarity Scores

Zhixin Zhou
School of CS & IT
RMIT University
VIC 3001, Australia
zhixin.zhou@rmit.edu.au

Xiuzhen Zhang
School of CS & IT
RMIT University
VIC 3001, Australia
xiuzhen.zhang@rmit.edu.au

Phil Vines
School of CS & IT
RMIT University
VIC 3001, Australia
phil.vines@rmit.edu.au

Abstract *Blog retrieval is a new and challenging task. Instead of retrieving individual documents, this task requires retrieving collections of documents, or blog posts. It has been shown recently that the federated model of using post entries as retrieval units is an effective approach to blog retrieval, where aggregation of similarity scores for posts to rank blogs plays an important role in the final ranking of blogs. In this paper, we explore two approaches of aggregation describing the depth and width of topical relevance relationship between post entries and blogs. We further propose holistic approaches that combine both approaches. Our experiments show that the sum baseline has the best performance, although the performances of the probabilistic approach and the linear pooling approach are very similar.*

Keywords blog retrieval, score aggregation

1 INTRODUCTION

The past decade has seen a surge of user-generated data on the web, among which the blogs play an important role. The term *blogosphere* refers to the whole collection of blogs on the Web.

A *blog* (also referred to as *web log*) is usually created and maintained by a web user who shares his or her writings on the web. Each *entry* of such writings is called a *blog post*, and is often followed by a list of replies from other web users, who contribute to the blog by adding their responses. Readers of blogs usually follow a subscription pattern, where they see an interesting blog post, browse through the other posts in the same blog, and if still interested, subscribe to the blog with a reader software which automatically tracks the updates through a file known as the *blog feed* (in the format of RSS[1] or ATOM[2]). One of the most popular instances of this kind of software is *Google Reader*[3].

A typical blog search behavior is shown in Figure 1, where *Jane Anderson* is a fictional blog author who frequently comments on brands of cosmetics. Many large companies also maintain official blogs for customer support or marketing purposes, among which is Google[4]. In the example given, a web user wishes to explore public opinions about Lancome, while another user would like to be informed about news from Google. After finding the blogs, the two users subscribe to them via feeds.

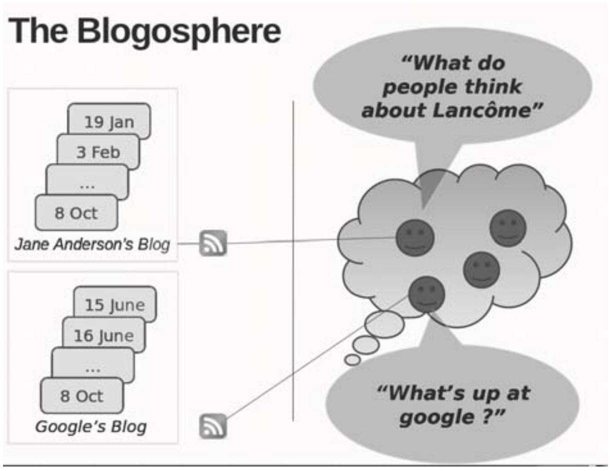


Figure 1: The blog searching behavior

Blog[5] retrieval, also known as *blog search* or *feed retrieval*, has many distinguishing characteristics from traditional document retrieval [5]. One of the essential differences is that the retrieval unit is no longer a single document, but a collection of documents (blog posts) to be evaluated as a whole. The TREC conference set up the Blog Track [13, 9, 12, 10] in 2007 to study search behaviors in the blogosphere, and introduced the Blog Distillation Task in 2007 to address the specific challenges posed by blog retrieval, also known as *blog feed search* [5]. The task is defined as to "*find me a blog with a principal, recurring interest in X*", where X is a topic of interest. Although a blog can be arguably viewed as a large virtual document comprising all posts to which

[1] http://en.wikipedia.org/wiki/RSS
[2] http://en.wikipedia.org/wiki/ATOM
[3] http://reader.google.com

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 4 December 2010.

[4] http://googleblog.blogspot.com/
[5] In this paper, we refer to blogs and feeds interchangeably as there is a one-to-one mapping between the two.

traditional retrieval techniques can be directly applied, recent work by Elsas et.al [5] has shown that a federated model that considers the topical relationship between a blog and its post entries is an effective approach to blog retrieval.

The aggregation of similarity scores for posts to rank blogs to establish the topic relevance relationship between posts and blogs and therefore plays an important role in effective blog retrieval. In this paper we explore aggregation approaches for combining both views of topical relevance relationship between blogs and their posts. Our problem can be described as follows. Existing document retrieval systems are able to estimate query relevance at the post level, and often in the form of similarity scores. Assuming that such scores are a reasonable measure of the post-level relevance, our research question can be stated as: *What is the best approach to aggregate post similarity scores for blogs so as to rank the blog feeds by query relevance?*

First we need to define what a *relevant blog* is. However, there is not a widely adopted definition of a relevant blog. According to the TREC definition of the blog distillation task, blogs that show ”*principal, recurring interest*” are the target of retrieval. In terms of the topic relevance relationship between blogs and their post entries this definition expresses the depth and width aspects of the topical relevance relationship between blogs and their post entries. In our discussions we use both sets of terms interchangeably.

Unfortunately, *principal* and *recurring* address different dimensions of the relevance. The *principal* dimension focuses on the relative percentage of relevant content in a blog, while the *recurring* dimension indicates the absolute amount of relevant content. Let us consider two blogs, one with 100 posts among which 20 were relevant, the other with 10 posts which are all relevant, and we assume that the relevant posts share the same degree of relevance. In this scenario the first blog shows more recurring interest as it has twice the number of relevant posts, whereas the second shows more principal interest since all of its posts were related to the topic.

A natural question is which one of the principal and recurring dimensions of relevance is more important for determining blog relevance to a topic. We study both aspects in terms of the topical relevance relationship between blogs and their posts. We also propose approaches combining the two aspects for ranking blogs.

The rest of this paper is organized as follows. In Section 2, we briefly survey related work on blog retrieval. In Section 3, we provide a detailed explanation of the approaches proposed. In Section 4, we present our framework for evaluation. In Section 5, we describe the setup of our experiments and discuss the results. Finally, the conclusion is made with an outlook on future works in Section 6.

2 RELATED WORK

The recent work of Elsas et.al [5] adapted a federated search model for blog retrieval. They showed that the federated model with blog posts as retrieval units outperformed the large-document model viewing a blog as a large documents comprising all posts. The focus of their work is on studying the pseudo-relevance feedback for posts and adapting it to improve blog retrieval. Our work is based on a similar blog retrieval model where post entries are the base retrieval units. We instead focus on how to aggregate the post relevance to achieve effective blog retrieval, which complements their study.

Blog search is a relatively new task. Related work started in 2007, when the blog distillation task was introduced into the TREC Blog Track [9, 12, 10]. Many participating groups approached this task by adapting techniques used in other existing search tasks. In this paper we focus only on major approaches used in post score aggregation, or in other words the document representation model.

He et al from the University of Glasgow [7, 6] compared the blog distillation task to the *expert finding* task of the Enterprise track [1]. Expert finding is the task of ranking candidate people as potential subject matter experts with respect to a given query [2].Each expert is associated with a collection of documents, and the retrieval model for this task assumes that expert candidates would have a large number of documents relevant to the query. This is similar to the blog retrieval task where each blog has a collection of blog posts, and a relevant blog would have a large number of relevant posts. He et.al adapted their Voting Model used in expert finding to feed search. Their model considers both the count of relevant posts in a feed and the extent to which each post is relevant.

Seo et al from the University of Massachusetts [15, 16] viewed this task as a *resource selection* problem, which originated from the distributed retrieval paradigm. Distributed information retrieval is also known as federated search, deals with document retrieval across a number of collections. The resource selection task aims to rank the document colletions so as to select the ones which are most likely to contain a large number of relevant documents. In their work, the geometric mean of the the query likelihoods of ”pseudo-clusters”, which are essentially the most relevant posts in a blog, was used to evaluate the blog’s relevance to the query.

Our work is different from the above described approaches in many ways. First, we focus our study on score aggregation and do not use any query expansion module or proximity search techinques. Second, we examine the relevance of blogs in two dimensions, which have not been addressed this way by previous works. Third, we have proposed holistic score aggregation approaches combing the two dimensions of post relevance, which are both shown to be effective.

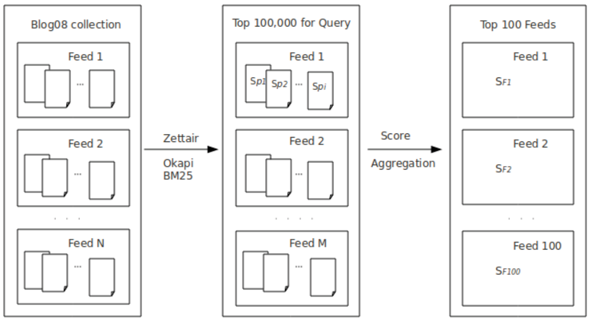


Figure 2: The Blog Retrieval Framework

3 THE BLOG RETRIEVAL FRAMEWORK

Our overall blog retrieval framework is illustrated in Figure 2. The framework comprises two components, namely, post retrieval and post aggregation. First we employ a document retrieval system to rank all blog posts in the collection by their estimated relevance to to a given query, and retrieve the top N. Ideally, to examine both the depth and width of relevance of a blog to a topic, the topical relevance of all post entries should be considered. However, given the size of the collection (the Blog08 collection we used has 28,488,766 unique posts), taking all posts into consideration is computationally costly. When N is sufficiently large, we could assume that the top N posts are a good representation of the collection with regard to the given topic, as the posts beyond this range are very unlikely to be relevant according to the estimation made by the retrieval system. To simulate the situation in our setting and also make the task manageable, for each topic, only the top 100,000 posts returned by the search engine were kept in the pool. In our experiments we also applied different thresholds to the post similarity scores, so that sub-collections with different levels of average query relevance are selected for investigation.

Theoretically any document retrieval model can be applied to retrieve blog posts. In our implementation we use the Zettair search engine [6] as the core document retrieval system. Zettair is a fast text search engine developed by the Search Engine Group at RMIT University. The Okapi BM25 model [8] is the core retrieval algorithm in Zettair, which is a probabilistic model. The similarity score of a document to a query, denoted as $S_{q,d}$, is an estimation of how closely the content of the document matches the query. The Okapi BM25 model makes use of statistical information about the distribution of the query terms (within both the document and the collection as a whole) to calculate the post similarity score.

$$S_{q,d} = \sum_{t \in \tau_{q,d}} w_t \times \frac{(k_1+1)f_{d,t}}{K+f_{d,t}} \times \frac{(k_3+1)f_{q,t}}{k_3+f_{q,t}} \quad (1)$$

The parameters in the equation are shown below,

q The query

d The document (blog post)

t A term in the query

w_t A representation of the inverse document frequency, calculated by $w_t = \ln \frac{N_d - N_{d_t} + 0.5}{N_{d_t} + 0.5}$ where N_d is the number of documents in the whole collection and N_{d_t} is the number of documents that contains the term t

$\tau_{q,d}$ The intersection of the distinct terms from the query and the document

k_1 A constant within the range of 1.2 to 1.5

k_3 A constant set to 1000000 (effectively infinite)

b A constant within the range of 0.6 and 0.75

K Calculated by $K = k_1 \times \left[(1-b) + \frac{b \cdot W_d}{W_{AL}}\right]$, where W_d is the document length and W_{AL} is the average document length in the collection

The last step of our blog retrieval framework is aggregating the query relevance for posts to score and rank blogs for topical relevance. This is a crucial step for the effectiveness of the whole system. Most existing systems estimate the query relevance in the form of similarity scores.These scores may or may not be distributed in a uniform manner [11], but it is usually possible to transform them into a uniform space. We consider the width and depth dimensions of the topical relevance of blogs, in terms of aggregating post similarity scores. We first discuss baseline approaches for aggregation in the next section and then propose two holistic approaches later.

4 BASELINE APPROACHES TO SCORE AGGREGATION

Corresponding to the width and depth dimensions of topic relevance of blogs we propose two baseline approaches aggregating post similarity scores to produce the blog relevance score. Given a blog F of n posts and post relevance scores $\{s_1, s_2, ..., s_n\}$, the blog similarity score S_F could be calculated from the post relevance scores $\{s_1, s_2, ..., s_n\}$,

The Average Baseline With the average baseline approach, the average post similarity score is used to estimate the query relevance of a blog, as shown in the equation below.

$$S_F = \frac{\sum_{i=1}^{n} s_i}{n} \quad (2)$$

where s_i is the similarity score of the i^{th} post.

This approach addresses the aspect of ”*principal interest*” as stated in the definition of the blog distillation task by TREC. It reveals the relationship between the

[6] http://www.seg.rmit.edu.au/zettair/

average degree of the query relevance of posts and that of the blog. Intuitively, blogs with a large number of posts are penalized. For instance, a blog F_A with 100 posts among which 10 were relevant should be considered more relevant than a blog F_B with 10 posts among which 5 were relevant. However, the average post relevance of F_A is likely to be lower than that of F_B.

The Sum Baseline With the sum baseline approach, the sum of post similarity scores is used to estimate the query relevance of a blog, as shown in the equation below.

$$S_F = \sum_{i=1}^{n} s_i \tag{3}$$

where s_i is the similarity score of the i^{th} post. This approach addresses the aspect of ”*recurring interest*” as stated in the definition of the blog distillation task by TREC. It reveals the relationship between the absolute amount of relevant content in a blog and its overall relevance to the blog. Intuitively, it discriminates against blogs that are specialized in one topic but having a low count of posts.

5 THE PROBABILISTIC MODEL

We propose to estimate the likeilihood of a blog's relevance to a given query from the degree of relevance of its post entries. In this approach we assume that a blog is considered irrelevant only if all posts in the blog are irrelevant. To calcluate the probability of the event that a blog be relevant to the query, we first transformed the post similarity scores in a feed F into probabilistic values,

$$p_i = \frac{s_i - s_{Q_{lower}}}{s_{Q_{upper}} - s_{Q_{lower}}} \tag{4}$$

where p_i is the probability of the i^{th} post being relevant to the query, Q_{upper} is the highest similarity score of all posts relevant to the query Q, Q_{lower} is the lowest similarity score of all posts relevant to the query Q, and s_i is the similarity score of the i^{th} post. We performed the transformation on a per-topic basis, as we expected different distributions of post similarity scores for each topic. Based on our assumption, the probability of the blog being irrelevant can then be calculated as,

$$\bar{P}_F = \prod_{i=1}^{n}(1 - p_i) \tag{5}$$

As a blog can be either relevant or irrelevant, the probability of a blog being relevant is thus,

$$P_F = 1 - \prod_{i=1}^{n}(1 - p_i) \tag{6}$$

Intuitively, this approach would not work well with blogs with a large count of irrelevant posts, as $\prod_{i=1}^{n}(1 - p_i)$ shrinks dramatically even if p_i is sufficiently small. We circumvent this problem by applying a threshold on p_i, so that only the ”relevant" posts are considered. Here, the probability of the blog being irrelevant is no longer the probability of all its posts being irrelevant. Instead, it is calculated as the probability of all ”relevant” posts in this feed being irrelevant, where the ”relevant” posts are selected by the threshold applied on the similarity score of the posts. Effectively this is setting the probability of low-score posts being relevant to zero. And since the similariy score is a reasonable indicator of the query relevance, the low-scored posts can be assumed to be irrelevant.

6 LINEAR POOLING: A HOLISTIC APPROACH

We propose an approach combining the depth and width dimensions of topical relevance for aggregating post similarity scores. As will be discussed in the Experiments section, the approach outperforms the baseline approaches.

Pooling distributions is a general approach to combining information from multiple sources or approaches, where sources are typically represented as probability distributions [4]. Here we consider two approaches, one based on the average baseline model and focusing on the width of relevance and the other based on the sum baseline model and focusing on the depth of relevance. We adopt the linear pooling approach to aggregating the estimations from these two approaches.

The distributions of scores over the two baselines are different. The scores from the average baseline range from 0 to 1, while those from the sum baseline can value above 100. Therefore we transform the feed scores into z-scores first, and combine the z-score value for the blogs. The z-score is calculated by the following formula:

$$s_Z = \frac{s - \mu}{\sigma} \tag{7}$$

where μ is the mean of all scores in the current distribution, and σ the standard deviation. Since the z-score measures the distance between a score and the mean score in the distribution and normalize that with the standard deviation, it allows scores form two different distributions to be comparable to each other.

We combine the two scores for each feed as follows:

$$P_F = \alpha * S_{F1} + (1 - \alpha) * S_{F2} \tag{8}$$

where S_{F1} and S_{F2} are the z-scores computed from the two probability values for blog relevance, calculated by Equation 4 for the average baseline model and the sum baseline model. Note that α and $1 - \alpha$ are the weights for the average baseline model and the sum baseline model respectively. By default we set $\alpha = 0.5$. Generally α can be adjusted to bias towards the depth dimension or the width dimension.

Table 1: Blog08 Collection Statistics (sourced from TREC Overview Paper[10])

Quantity	Blog08
Number of Unique Blogs	1,303,520
First Feed Crawl	14/01/2008
Last Feed Crawl	10/02/2009
Number of Permalinks	28,488,766
Total Compressed Size	453GB
Total Uncompressed Size	2309GB
Feeds (Uncompressed)	808GB
Permalinks (Uncompressed)	1445GB
Homepages (Uncompressed)	56GB

7 EXPERIMENTS

7.1 Dataset

Our experiments were done on the Blog08 collection used for TREC 2009 and TREC 2010 Blog Track conferences. This collection was created by the University of Glasgow to provide an experimental environment for the Blog Track. Summary statistics for this collection are listed in Table 1.

The collection contains three types of data, namely, permalinks (blog posts), feeds, and homepages. We only indexed the permalinks in our experiments. Each permalink document is associated wth one feed, whereas a feed could be associated with multiple permalink documents. On average, each blog contains 22 posts in our collection.

We tested our approaches with the 49 topics used in TREC 2009 Blog Track. We used only the topic title text as our queries, which are typically short expressions comprising of two or three words such as ”genealogical sources” (topic 1101). The query relevance judgments were done by NIST, against which the estimations made by our blog retrieval system were evaluated.

7.2 Evaluation

We follow the evaluation methods adopted by the Text REtrieval Conference [14, 3]. Four metrics were used, namely, MAP, P@10, R-prec, and B-pref. Each of these metrics address a different aspect of the performance of the retrieval system, where the MAP is the main measure of the system's performance. This is also the main measure used in the TREC Blog Track. The results we show are generated by the trec_eval [7] software provided by the TREC conference.

MAP Precision and recall are single-value metrics based on the whole list of documents returned by the system. For systems that return a ranked sequence of documents, it is desirable to also consider the order in which the returned documents are presented. MAP, or Mean Average Precision, emphasizes ranking relevant documents higher. It is the average of precisions computed at the point of each of the relevant documents in the ranked sequence.

P@10 P@10 (Precision at 10 documents) counts the number of relevant documents in the top 10 documents in the ranked list returned for a topic. This precision correlates with the precision observed by a web user.

R-prec R-prec is the precision computed after R documents have been retrieved, where R is the number of relevant documents for the topic. Contrary to MAP, this metric de-emphasizes the exact ranking of the retrieved relevant documents.

B-pref B-pref is robust in collections which may have incomplete relevance information. The idea is to measure the effectiveness of a system on the basis of judged documents only. R-precision, MAP, and P(10) are completely determined by the ranks of the relevant documents in the result set, and make no distinction in pooled collections between documents that are explicitly judged as nonrelevant and documents that are assumed to be nonrelevant because they are unjudged. B-pref, on the other hand, is a function of the number of times judged non-relevant documents are retrieved before relevant documents.

We applied thresholds on the similarity scores to select different collections of posts with different levels of average relevance. First we scaled the similarity score of each post to a probability value between [0,1], with the method defined in Equation 4. Different threshold settings on these probability scores allow us to examine the performance of our approaches in different post collections in terms of average query relevance. For each topic, we applied 9 thresholds, namely 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, to each approach except the linear pooling approach. This is because with that approach, a majority of the topics do not have any post with a score above 0.8.

7.3 Results and Discussion

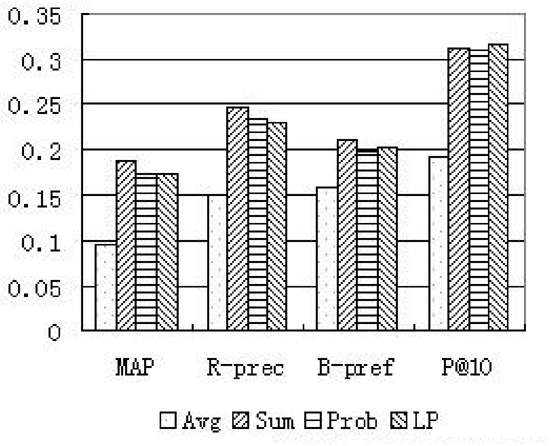


Figure 3: Overview

[7] http://trec.nist.gov/trec_eval/index.html

Figure 3 provides an overview of the approaches we used. The run label *Avg* corresponds to the average baseline, label *Sum* refers to the sum baseline, label *Prob* is for the probabilistic approach, and label *LP* refers to the linear pooling approach. As is shown in the graph, the sum baseline outperformed all other approaches when evaluated with all metrics used except P@10. The probabilistic approach and the linear pooling approach have similar performance to that of the sum baseline, but the average baseline is significantly worse than the other approaches. Note that all the data shown in this graph was not obtained under the same threshold setting. Instead, the best run for each approach was selected from a number of runs under different settings. The effect of different thresholds is discussed below.

We extracted individual topic performance by MAP for each approach, and used the paired Wilcoxon test to compare the difference between the approaches. The performance of the sum baseline, the probabilistic approach and the linear pooling approach were significantly better than that of the average baseline, with $p < 1.449e-6$, $p < 9.942e-5$, $p < 4.7e-6$ respectively. The performance of the sum baseline is significantly better than the linear pooling approach as well, with $p < 0.005184$, but the difference between its performance and that of the probabilistic approach is insignificant. There is no significant difference between the performance of the probabilistic approach and the linear pooling approach, either.

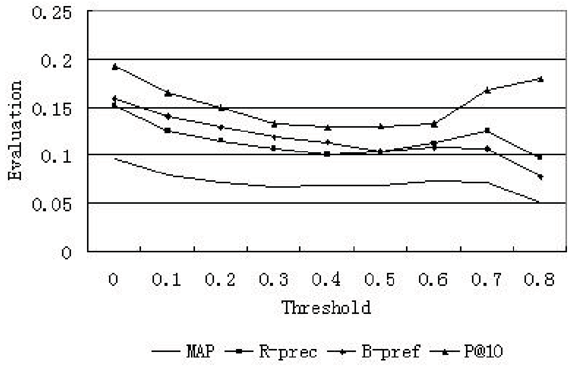


Figure 4: The Average Baseline

Figure 4 shows the performance of the average baseline with different thresholds applied to the post scores. With metrics other than b-pref, the performance deterioriates as the threshold gets higher, but improves slightly near the threshold 0.7. Above that threshold, the performance again declines with all metrics but P@10. This is proabably due to the fact that only a small number of blogs have posts with such a high score, and with such highly relevant posts they are very likely to be relevant. With P@10, only the top 10 blogs are evaluated, and is not affected by the decrease in the number of feeds found. The improvement over the performance reflected by P@10 also implies that highly relevant posts suggests a high blog relevance, but there is a tradeoff between the precision and the recall, as is reflected by the other metrics.

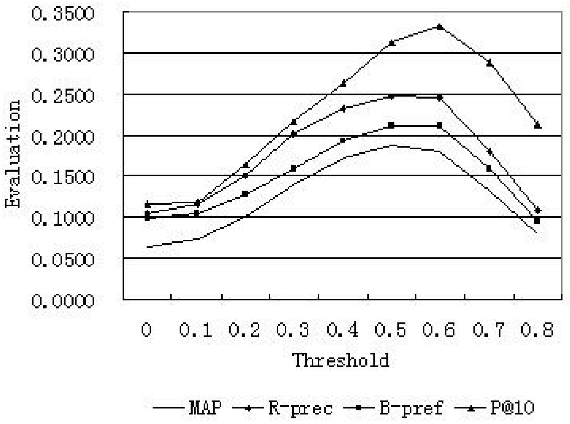


Figure 5: The Sum Baseline

Figure 5 shows the performance of the sum baseline under different thresholds. The trend shown in the graph is consistent with all metrics we used. The performance of the sum baseline improves as the thresholds becomes higher, and peaks near 0.5 and 0.6. After that, the performance declines, due to a decrease in the number of feeds found. This is in accordance with our observation with the average baseline. It also supports the implication that a group of highly relevant posts in a blog determines the query relevance of the blog.

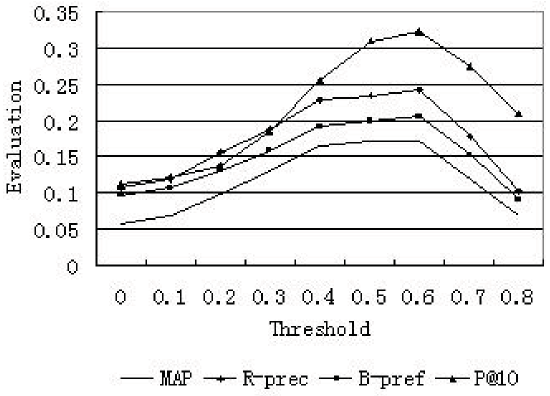


Figure 6: The Probabilistic Approach

Figure 6 shows the performance of the probabilistic model we proposed. Overall, the performance of this approach is very similar to that of the sum baseline. When evaluated with MAP, the performance of the sum approach peaked at 0.5, while the performance of the probabilistic model peaked at 0.6, but the difference between their performance at 0.5 and 0.6 was negligible.

Figure 7 and 8 shows the performance of the linear pooling approach we proposed. As is shown in Figure 7, the performance of this approach is similar to that of the sum baseline and the probabilistic model. It is worth noting however, that this approach has achieved the best

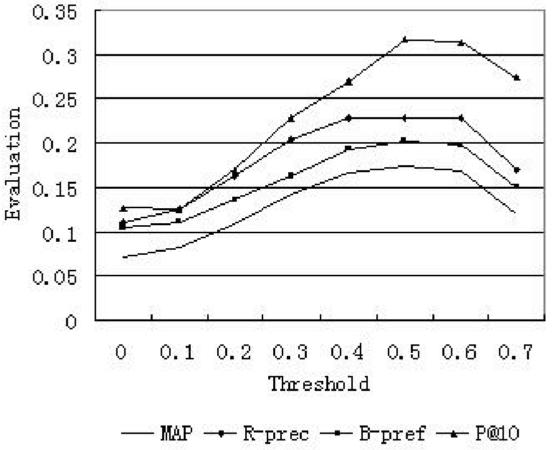


Figure 7: The Linear Pooling Approach

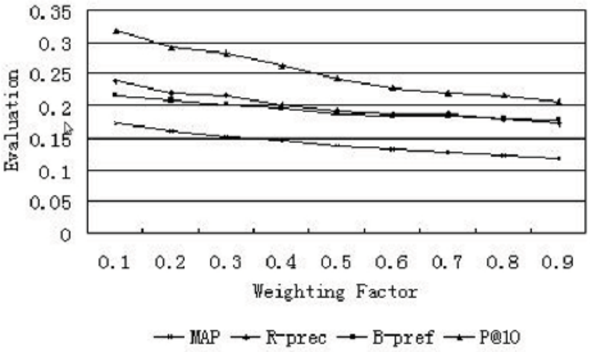


Figure 8: The Weigthing Factor α

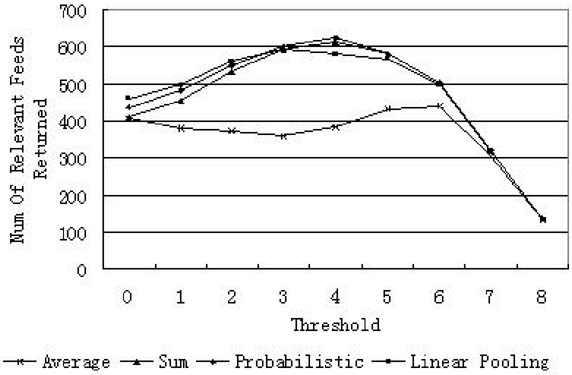


Figure 9: Num of judged relevant feeds returned

performance among all approaches when evaluated by P@10. Figure 8 shows how the weighting factor α (defined in Equation 8) influences the performance of the linear approach. In this graph we aggregated two baselines with different values for α. The two baselines we used were the average baseline with thresholds set to 0, and the sum baseline with thresholds set to 0.5. The choice of the thresholds was based on the performance of the two baselines we observed in our experiments, and we chose the ones which lead to the best evaluation results. However, as we have not tested the values exhaustively, the values we chose may not be optimal. From the graph we can see that the smaller the weighting factor is, the better performance appears to be. As a smaller α suggests larger weight for the sum baseline, and the best performance observed when $\alpha = 0.1$ is still inferior to that of the sum baseline, the graph implies that the average amount of the post relevance, or *the principal interest*, in a feed may not be as important as the total amount of the post relevance (taking only posts with a relatively high relevance score into account), or the *recurring interest*.

Overall, with all approaches other than the average baseline, the performance of the system is positively influenced by higher threshold settings, although it will be hurt by a drop in recall when the threshold rises to above 0.6, which is shown in 9. This observation implies that the posts with a high query relevance in a feed determines the blog relevance.

Interestingly, the performance of the average baseline actually declines with higher thresholds, although it rose a little when we used only posts with a score above 0.6. It is also worth noting that when using all posts (when threshold is 0), the average baseline has the best performance. Combined with our implication from the other three approaches, we deduce that the weights that the two aspect of the blog relevance carry vary under different threshold settings. While the average post relevance is a reasonable indicator of the blog relevance, the potentially huge amount of irrelevant posts could greatly hurt its viability. This also explains why the sum baseline has an extremely poor performance when using all posts (the accumulation of the low scores favors blogs with a large amount of posts). On the other hand, when considering two blogs A and B, both containing some highly relevant posts, and assuming that blog A has a larger count of highly relevant posts whereas blog B has a higher average post relevance, blog A is probably more relevant than blog B. This explains why elevating the threshold hurts the performance of the average baseline.

8 CONCLUSIONS AND FUTURE WORK

In this paper we have explored a number of approaches to estimate the query relevance of blogs. First we examined two baseline approaches based on the definition of the blog retrieval task, each addressing a different dimension of the blog relevance, namely, *principal* and *recurring*. Our experiment results show that highly relevant posts are a good representation of the feed in terms of its topical relevance. Our result also suggests that despite the fact that relevance judgment is affected by both the average degree of the query similarity and the number of relevant retrieval units, it is not sensitive to the former so long as a threshold is met.

We also proposed two holistic approaches which address both dimensions of the blog relevance. The two approaches have similar performances, both better than the average baseline but very close to that of the sum baseline.

This work focused only on the score aggregation techniques. To further improve the performance of the approaches we proposed, we also plan to examine the techniques for score normalization. More flexible settings of thresholds based on the distribution of post scores within a blog are also subject to further study.

References

[1] P. Bailey, N. Craswell, A. P de Vries and I. Soboroff. Overview of the TREC 2007 enterprise track. In *Proceedings of TREC*, 2007.

[2] K. Balog, L. Azzopardi and M. de Rijke. Formal models for expert finding in enterprise corpora. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, page 4350, 2006.

[3] C. Buckley and E. M Voorhees. Retrieval evaluation with incomplete information. In *Proceedings of the 27th annual internationa ACM SIGIR conference on Research and development in information retrieval*, 2004.

[4] R. T Clemen and R. L Winkler. Combining probability distributions from experts in risk analysis. *Risk Analysis*, Volume 19, Number 2, pages 187203, 1999.

[5] J. L Elsas, J. Arguello, J. Callan and J. G Carbonell. Retrieval and feedback models for blog feed search. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, page 347354, 2008.

[6] D. Hannah, C. Macdonald, J. Peng, B. He and I. Ounis. University of glasgow at TREC 2007: Experiments in blog and enterprise tracks with terrier. In *Proceedings of TREC*, Volume 2008, 2007.

[7] B. He, C. Macdonald, I. Ounis, J. Peng, R. L Santos and GLASGOW UNIV (UNITED KINGDOM). University of glasgow at TREC 2008: experiments in blog, enterprise, and relevance feedback tracks with terrier. In *Proceedings of TREC*, 2008.

[8] K. Sparck Jones, S. Walker and S. E Robertson. A probabilistic model of information retrieval: development and comparative experiments:: Part 2. *Information Processing & Management*, Volume 36, Number 6, pages 809840, 2000.

[9] C. Macdonald, I. Ounis and I. Soboroff. Overview of the TREC 2007 blog track. In *Proceedings of TREC 2007*, 2007.

[10] C. Macdonald, I. Ounis and I. Soboroff. Overview of the TREC 2009 blog track. In *Proceedings of TREC 2009*, 2010.

[11] R. Manmatha, T. Rath and F. Feng. 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*, page 267275, 2001.

[12] I. Ounis, C. Macdonald, I. Soboroff and GLASGOW UNIV (UNITED KINGDOM). Overview of the TREC 2008 blog track. In *Overview of the TREC 2008 blog track*, 2008.

[13] I. Ounis, M. De Rijke, C. Macdonald, G. Mishne and I. Soboroff. Overview of the TREC-2006 blog track. In *Proceedings of TREC*, Volume 6, 2006.

[14] T. Sakai. Comparing metrics across trec and ntcir: the robustness to system bias. In *Proceedings of the 17th ACM conference on Information and knowledge management*, 2008.

[15] J. Seo and W. B Croft. Umass at trec 2007 blog distillation task. In *Proc. of the 2007 Text Retrieval Conf*, 2007.

[16] J. Seo and W. B Croft. Blog site search using resource selection. In *Proceeding of the 17th ACM conference on Information and knowledge management*, page 10531062, 2008.

Estimating System Effectiveness Scores With Incomplete Evidence

Sri Devi Ravana[1,2] *Alistair Moffat*[1]

1. Department of Computer Science and Software Engineering
The University of Melbourne
Victoria 3010, Australia

2. Department of Information Science
University of Malaya
Kuala Lumpur 50603, Malaysia

{*sravana,alistair*}*@csse.unimelb.edu.au*

Abstract *It is common for only partial relevance judgments to be used when comparing retrieval system effectiveness, in order to control experimental cost. Using TREC data, we consider the uncertainty introduced into per-topic effectiveness scores by pooled judgments, and measure the effect that incomplete evidence has on both the systems scores that are generated, and also on the quality of paired system comparisons. We measure system behavior from three different points of view: the trend in effectiveness scores; the separability of system pairs; and the number of reversals in significance outcomes as the depth of judgments increases. Our results show that when shallow pooled judgments are used system separability remains relatively high, but that there is also a high rate of significance reversal. We then show that explicitly adjusting effectiveness scores to allow for the known amount of uncertainty gives a reduced number of reversals, and hence more consistent experimental outcomes.*

Keywords Retrieval evaluation, effectiveness metric, pooling

1 Introduction

It is now nearly twenty years since TREC-style large-scale experimentation comparing retrieval techniques was commenced. One facet of such experiments that has remained constant over these two decades is the tension between the cost of undertaking relevance judgments, and the desire for accuracy of measurement. An experiment can be relatively low-cost if only shallow judgments are undertaken, but that then means that "deep" effectiveness metrics (such as average precision, AP, and normalized discounted cumulative gain, NDCG) cannot be properly computed. As a result, a range of work has been undertaken to quantify the extent to which the values of deep metrics are correlated to shallow metrics, see, for example, Webber et al. [22].

In this paper we take a different approach, and seek to quantify the extent to which system pair comparisons are inaccurate when only shallow judgments are performed. In particular, we make use of TREC experimental runs and TREC relevance judgments to investigate whether pairwise relativities that are deemed significant when only shallow judgments are available remain significant when deeper judgments are provided, for a range of effectiveness metrics. We call this the *reversal rate* of an experiment – the extent to which the use of shallow judgments leads to conclusions of statistical significance that are not in fact supported when a fuller set of relevance judgments is used in the calculation of the effectiveness metric.

The results presented below show that the usual simplistic pooling assumption – that documents that are unjudged are irrelevant – leads to a higher reversal rate than methods that attempt to infer effectiveness scores based on other assumptions about unjudged documents. This outcome suggests that if shallow pooling is being used during an experiment, an appropriate mechanism for estimating effectiveness scores should also be employed.

2 Retrieval experimentation

Retrieval systems are often evaluated using prescribed test collections, fixed topic sets, and matching relevance judgments. This is particularly true in non-commercial research environments, in which access to query and click logs, and to other large-scale user interaction data, is limited by competition or privacy concerns. But formation of relevance judgments is costly, and so it is also usual for the relevance judgment sets to be generated once in a shared effort, and then reused as ground-truth by subsequent experimentation. This section briefly summarizes this type of collection-based retrieval experimentation, and outlines the various facets of the process that have been subject to scrutiny.

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

Collections and topics The first step is to compile a suitable document collection. Among others, the annual TREC rounds have used newswire collections, government web pages, and patent repositories. Topics have been based on a range of statements of information need, matched to the collection. For details of these aspects of experimental design, see Voorhees and Harman [20].

Pooling Collection-based testing had its origins in the Cranfield collection about 40 years ago [2, 20, 5] which consists of about 1,400 abstracts and 225 requests. At that scale it was possible to be confident that all of the relevant documents were known, and that the relevance judgments were *complete*. With the use of larger (and hence more realistic) collections, it is impractical to generate complete judgments [17]. Instead, the documents are triaged into three sets in regard to each topic: those that have been judged and are relevant; those that have been judged and are irrelevant; and those that have not been judged.

To select the subset of the documents that will be judged for each topic, *pooling* is used [15]. To form a pool for each topic, each system in the set of s participating systems ranks the documents in relation to that topic. These s ranked lists are then truncated to some fixed *pool depth* d, and the list prefixes are combined and de-duplicated. This process focusses the judgment effort on at most sd documents, and means that, as a minimum, in each of the s system runs, the highest ranked d documents have all been judged. Presuming that each of the s systems prioritizes its ranking on the documents perceived as being most likely to be relevant, the overall set of judgments is similarly focussed on the documents that are most likely to be relevant. Test collections developed using this technique have been investigated in a number of ways and found to be relatively reliable in terms of their ability to predict system behavior on unknown topics [19, 26].

In contrast to these earlier evaluations, recent work has suggested that even pooling cannot completely eliminate the need to assess significant numbers of documents when large collections are being used, because the fraction of documents pooled compared to the collection size is small. In these cases the results generated using these relevance judgments may not be reliable [4].

Related work Many new methods and techniques have been introduced in recent years that seek to overcome the problems generated by incomplete judgments, including: alternative strategies that seek to increase the number of relevant documents identified [26], or otherwise adjust the order in which documents are added into a queue for judgment [9]; the use of multiple assessors per topic [16]; incorporating prior system scores into extended experiments [11]; reducing judging effort while maintaining a large number of topics [5]; identifying topic difficulty to provide reliable results [25]; evaluation without relevance judgments [1, 24]; and score adjustment for pooling bias [21]. Ali et al. [1], Sanderson and Zobel [14], Trotman and Jenkinson [16], and Voorhees [18] examine other aspects of test collection construction, and of pooling as a technique for identifying documents to be judged.

Effectiveness metrics Closely coupled with the issue of pooling is the question of which effectiveness metric should be used. Shallow metrics, such as precision at depth k (P@k, with k often chosen to be a small number such as 10) are completely determined provided d, the pool depth, is chosen such that $d \geq k$. Finite-depth metrics of this type do not include any normalization factor that scales them against the best that any system might do on this topic; this absence means that the scores generated are absolute values. When $k > d$, the extent of the uncertainty in any P@k score can also be exactly known, since the possible contribution of each unjudged document is exactly $1/k$. This uncertainty is denoted as a *residual*, and represents the magnitude of the range in which the P@k score might ultimately sit. This notion of residuals is taken up in more detail in the next section.

On the other hand, deeper "system" metrics such as average precision (AP) [2] and normalized discounted cumulative gain (NDCG) [7] give rise to normalized scores that are scaled against the best that any system might achieve, with a "perfect" ranking always attaining a score of 1.0, regardless of how many relevant documents there are for the query. For these metrics to be correctly computed, the number R of relevant documents for each topic must be known, with the implication that any pooling-based approach to depth d will identify an approximation $R_d \leq R$ of that number.

As a compromise between these classes of metric, rank-biased precision (RBP) [8] computes an absolute score rather than a relative score, over any finite prefix of a presumed-infinite ranked list. Because only a finite prefix is ever scored, and because there is no normalization by R, it is again possible to compute a residual that indicates the extent of the uncertainty generated by the truncated tail of the ranking, or by any other unjudged documents within the supplied prefix.

Other effectiveness metrics have also been proposed, including ones that expressly seek to ameliorate the problems cased by incomplete judgments [3, 12, 13]. We do not consider these approaches further in this work; rather, we seek to apply score estimation techniques to the more traditional effectiveness metrics. In particular, we consider the problem of uncertainty introduced into per-system per-topic effectiveness scores when incomplete judgments are used in experiments. In our work we present results of investigating pairwise comparison of systems when estimating system scores in face of incomplete evidence.

3 Score Estimation

As has already been introduced, suppose that d is the pool depth used in the development of a set of relevance judgments, and that k is the run depth for some system that contributed to the pool. Then, as an example, P@k assesses that fraction of the top k ranked documents for each system that are relevant for each topic. When $k > d$ there are three sets of documents identified by this process:

- those judged relevant, r in total;
- those judged irrelevant, n in total; and
- those not judged, $k - (r + n)$ in total.

The P@k score for this system on this topic can then be expressed as the interval $[B, T]$, where $B = r/k$ and $T = 1 - n/k$ are the lower and upper bound on the P@k score. The P@k residual is then defined as $\Delta = T - B = 1 - (r + n)/k$. Computation of a residual for RBP and other weighted-precision metrics (including DCG, the unnormalized version of NDCG) is only a little more complex.

On the other hand, when metrics such as NDCG and AP are being used, the unjudged tail of any ranking can (at least potentially) dominate the score established by any finite prefix and it is not possible to establish a range for the score. Indeed, with these metrics, all that can be said is that in a pathological situation, the eventual score calculated for any document ranking lies between $B = 0$ and $T = 1$. More specifically, if the judgment depth d is extended to a new value $d' > d$, then computed AP and NDCG scores might increase or might decrease, whereas P@k, RBP, and DCG scores can only increase.

In practice, use of score intervals is unwieldy, and scores ranges are represented by a *point estimate* that is computed as some function of the available information, with the estimate X associated with a range $[B, T]$ required to satisfy $B \leq X \leq T$. Taking as a starting point the work of Ravana and Moffat [10], we explore four different estimation techniques in the experiments discussed below.

Simplistic prediction The simplest method is to take the lower bound of the interval, as the score estimate X,

$$X_S = B\,.$$

This is the "conventional" way of dealing with judgment uncertainties, and is best summarized as "if it ain't judged, it ain't relevant".

Background prediction A second option is to make use of a global estimate E that represents the background probability of a document being relevant given that it has been retrieved. The score associated with a $[B, T]$ interval can then be estimated as

$$X_B = B + \Delta E\,.$$

In this method, a fixed fraction of Δ is uniformly added to B. The value E is a constant and it can take any value from 0 to 1 although through experiments we observed that the $0.01 \leq E \leq 0.05$ is a reasonable range [10].

Interpolated prediction Assuming that the unjudged documents for a system are – to within some constant factor C – as likely to be relevant as the documents for which judgments are available leads to interpolated scores to be computed as:

$$X_I = B + C\Delta \frac{B}{1 - \Delta}\,.$$

Constant C is a value between 0 to 1, and suitable values are discussed shortly. A value for X_I cannot be computed when $\Delta = 1$ (that is, when $B = 0$ and $T = 1$), and in this special case $X_I = E$ is assumed.

Smoothed prediction Assuming that the lower the uncertainty Δ, the greater the confidence is in the Interpolated prediction, and in contrast the higher the uncertainty, the more the background model should be preferred, leads to a smoothed approach Ravana and Moffat [10]:

$$\alpha X_I + (1 - \alpha) X_B\,,$$

where α is a parameter that reflects the level of confidence in the Interpolated prediction. If α is chosen to be $\alpha = 1 - \Delta$, this simplifies to

$$X_M = B + C\Delta B + \Delta^2 E\,,$$

where, as before, C is a constant between zero and one.

Computing score ranges We experimented with a total of five different effectiveness metrics: P@10, a typical shallow metric; AP, the average precision when evaluated relative to R_d, where R_d is the number of relevant documents encountered in the first d items of any of the pooled system runs; NDCG, normalized cumulative discounted gain, again using R_d; SDCG at depth $k = 100$, the discounted cumulative gain (see Järvelin and Kekäläinen [7]) scaled by the maximum possible score possible at depth 100; and RBP, rank-biased precision, with parameter $p = 0.95$.

With all of P@10, SDCG, and RBP, the base value B and top value T that bookend each score interval are relatively straightforward to compute. With AP and NDCG neither B nor T is easy to compute, and instead we use an approximation to gauge the breadth of the $[B, T]$ interval. To establish a lower estimate B, the usual approach to computing the scores was followed, making the assumption that none of the unjudged documents that appeared in the ranking were relevant. As already noted, this is not a strict lower bound on the eventual score.

To calculate an upper estimate T of the score range, the number of judged relevant documents in the run was subtracted from R, the total number of relevant documents for the topic. The remaining relevant documents not already accounted for in the ranking were then assumed to be inserted into the ranking at the earliest possible locations at which unjudged documents appeared. For example, with $R = 5$ and $k = 10$, the ranking

```
1 0 ? 0 1 1 0 ? 0 ?
```

(in which "0" is an irrelevant document, "1" is a relevant document, and "?" is an unjudged document) gives rise to an AP-based $[B, T]$ interval computed as

$$B = \frac{1}{5}\left(\frac{1}{1} + \frac{2}{5} + \frac{3}{6}\right) = 0.38\,,$$

which is the usual computation with two terms completely absent; and

$$T = \frac{1}{5}\left(\frac{1}{1} + \frac{2}{3} + \frac{3}{5} + \frac{4}{6} + \frac{5}{8}\right) = 0.71\,,$$

now with those two terms inserted at the first available locations.

While this arrangement is in fact not feasible (since, in the example, it is known that neither of the two relevant documents that are absent from the ranking appear in positions 3 and 8), in conjunction with the B value, this method of estimating an approximation of T does give reasonable guidance as to the level of imprecision in the computed AP score. In particular, when $\Delta = T - B$ is large, then the B score may not be a good estimate of the final AP score. A similar approach allows estimates of B and T to be made for NDCG.

4 Experimental Investigation

Test Data We make extensive use of the relevance judgments and submitted runs that were generated during the TREC9 Web Track undertaken in 2000 [6]. This track had 105 runs submitted in response to a set of 50 topics. Of these, 59 runs were used in the pooling stage during which the set of judgments was generated, and 46 of the runs were not. From the 59 contributing runs (the set denoted as "59-con"), for each topic, the top 100 document identifiers from each run were pooled and judged, following the standard TREC methodology of generating relatively deep judgment sets.

The complete set of judgments for the TREC9 Web track contains 69,100 recorded outcomes, which means that on average each judged document got nominated by 4.3 of the 59 contributing systems. To generate simulations of shallower pools for experimental purposes, each judgment in the full set was tagged with the minimum depth at which that document was located in any of the 59-con runs, and then the judgments sorted into increasing order of minimum encountered depth. Prefixes of length 1,000 and 10,000 judgments were then taken, as a simulation of the outcomes that would arise if shallow and medium judgments were used.

The division into contributing (set 59-con) and non-contributing (set 46-non) runs is a useful one, and we report results separately for the two sets of systems. In particular, use of the 46-non set of system runs allows exploration of issues that arise when systems are being compared without any of them having contributed to the judgment set.

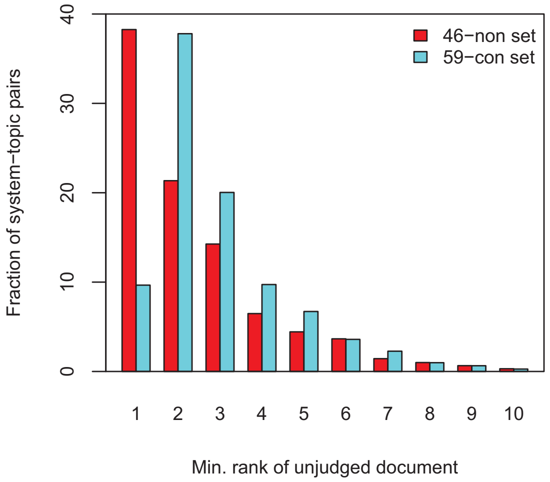


Figure 1: Distribution of ranks of first unjudged document in each run, categorized by whether or not the run contributed to the pool. A total of 1,000 judgments is assumed.

Figure 1 highlights this distinction. It shows, averaged across the systems and topics, the depth of the first unjudged document in each run when only 1,000 judgments are used. With this number of judgments to be distributed across 59 systems and 50 topics, the majority of runs in the 59-con set have their top-ranked document judged, but not always the second, so the effective pool depth is around $d = 1$. On the other hand, in the 46-non set, more than a third of the runs do not even get their top-ranked document judged.

Shallow pooling and uncertainty The first phase in our evaluation was to simply score the sets of runs using the three judgment sets, but measuring the extent of the uncertainty generated by the incomplete judgments. Table 1(a) shows the average base scores B computed for the 46-non systems, using five different effectiveness metrics, and evaluated using shallow, medium, and deep pooled judgments. In the case of the three weighted precision (and hence score accretive) metrics P@10, SDCG and RBP, the use of the $X_S = B$ approximation leads to non-decreasing score estimates as the number of judgments increases. On the other hand, the base AP score estimates decrease as the pool depth increases. This behavior is a consequence of R increasing, but those additional relevant documents not appearing in the majority (or even any) of the runs actually being scored. In between is NDCG, where it appears that the base score estimate B is relatively stable even from very shallow pool depths.

Table 1(b) lists the average residuals Δ associated with those base scores. The best that can be said about these values is that for the three weighted-precision metrics they decrease as the judgment pool increases in size. But as a general indication of scoring certainty, they provide very weak evidence. In particular, even

Judgments	P@10	SDCG	RBP	AP	NDCG
1,000	0.1184	0.0419	0.0647	0.1822	0.3282
10,000	0.1877	0.0900	0.1269	0.1541	0.3494
69,100	0.1923	0.1085	0.1398	0.1258	0.3237
(a) Base effectiveness scores, $X_S = B$					
1,000	0.6759	0.8535	0.7921	0.2744	0.2903
10,000	0.2692	0.5670	0.4333	0.2789	0.3240
69,100	0.1631	0.2311	0.1955	0.2309	0.3229
(b) Residuals resulting from unjudged documents, Δ					
1,000	0.1840	0.1373	0.1556	n/a	n/a
10,000	0.2019	0.1293	0.1566	n/a	n/a
69,100	0.1974	0.1157	0.1460	n/a	n/a
(c) Interpolated scores, X_I with $C = 0.42$ and $E = 0.01$					
1,000	0.1823	0.0995	0.1255	n/a	n/a
10,000	0.2064	0.1296	0.1589	n/a	n/a
69,100	0.2020	0.1216	0.1509	n/a	n/a
(d) Smoothed scores, X_M with $C = 0.91$ and $E = 0.05$					

Table 1: Base effectiveness scores B; residuals Δ; and two point estimates within the $[B,T]$ range, in all cases averaged across 50 topics and the 46-non set of system runs.

when the full set of 69,100 judgments is applied to the most focussed of the five metrics, P@10, not even one decimal digit of accuracy can be relied on. This clearly suggests that the simplistic point values X_S may not be accurate. (When the 59-con set of systems is used with $d = 100$ relevance judgments, the average residual for P@10 and SDCG@100 is zero, see Ravana and Moffat [10] for these and related results.)

The approximated residuals for the two normalized metrics, AP and NDCG , are both very large; nor do they decrease as the judgment pool increases in size. This suggests that either AP and NDCG scores are intrinsicly imprecise, or that the estimation methodology is inaccurate. Further work is required to determine which explanation is the correct one.

RMS error To quantify the difference between true and estimated values, we computed root-mean-square (RMS) errors. If Y is a set of n "true" values, $Y = [y_1, y_2, \ldots, y_n]$, and X is a corresponding set of estimated values, $X = [x_1, x_2, \ldots, x_n]$, then the root-mean-square difference between Y and X is computed as:

$$RMSE(X,Y) = \sqrt{\frac{\sum_{i=1}^{n}(x_i - y_i)^2}{n}}.$$

The smaller the RMSE value the better the predictive quality of the estimation method.

To determine constants C and E to be used in the Interpolative and Smoothed predictions methods X_I and X_M respectively, we took the set Y to be the 50×59 at-69,100 system-topic scores achieved by the 59-con systems. The set of estimates X was then computed for each system and each topic, based on six different pool depths of, variously, 1,000, 2,000, 4,000, 10,000, 20,000, and 40,000 judgments.

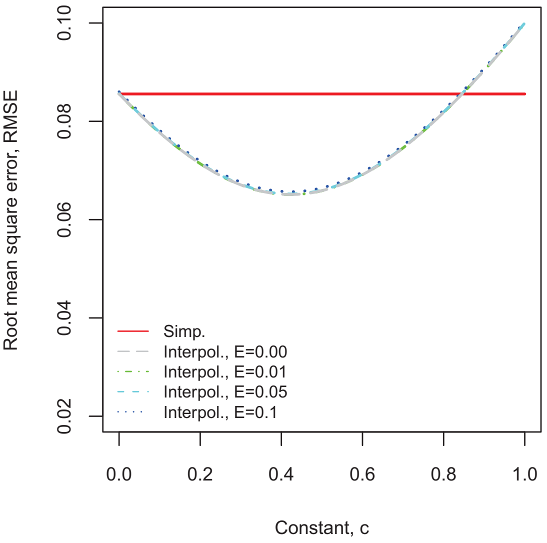


Figure 2: Prediction quality of P@10 scores estimated using the X_I Interpolated approach, plotted as RMSE values. The 59-con set was used with C and E varying, with results aggregated over six different pooling depths. The minimum point arises with $C = 0.42$ and $E = 0.01$.

Figures 2 and 3 show how *RMSE* varies as C and E are altered, using the Interpolated method to predict scores from $[B,T]$ ranges. The minimal X_I-RMSE value is 0.065, while the minimum X_M-RMSE is 0.086, achieved with different C and E values for the two different methods. Both of these two figures were generated using the metric P@10; broadly similar curves resulted for the SDCG and RBP metrics.

For the purpose of the experimentation, $C = 0.42$ and $E = 0.01$ are used in the Interpolated method X_I; and $C = 0.91$ and $E = 0.05$ are used in the Smoothed method X_M. Both combinations give smaller RMSE values than the simplistic predictor X_S, for all of P@10, SDCG, and RBP.

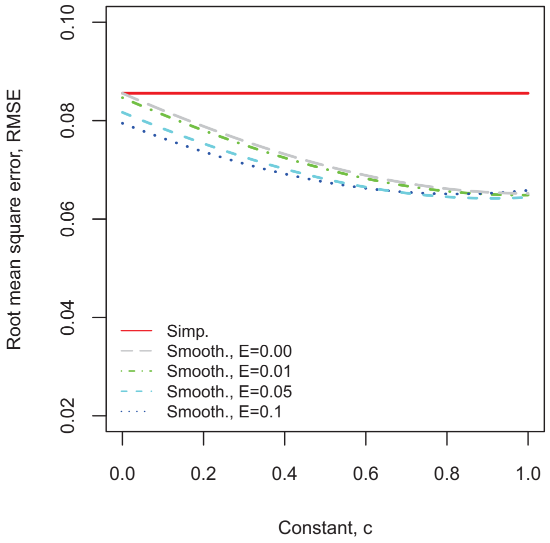


Figure 3: Prediction quality of P@10 scores estimated using the X_M Smoothed approach, plotted as RMSE values. The 59-con set was used with C and E varying, with results aggregated over six different pooling depths. The minimum point arises with $C = 0.91$ and $E = 0.05$.

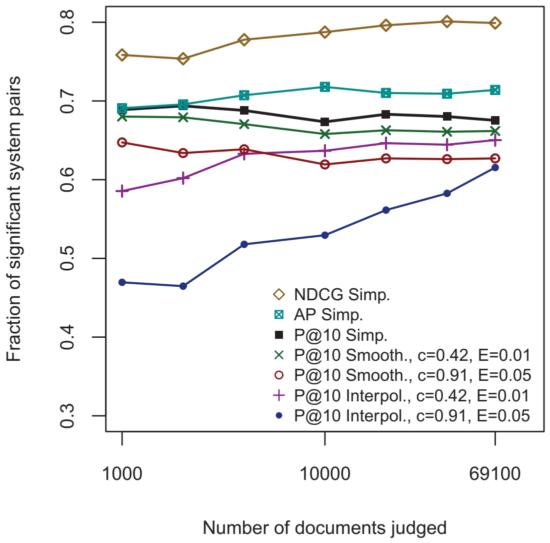


Figure 4: Separability rates within the 46-non set of systems for P@10 as a function of the total number of relevance judgments performed, with pooling across 50 topics and 59 systems, and with the comparison based on use of the t-test at the 0.01 confidence level.

Trends in effectiveness scores Table 1(c) applies these learnt constants to the 46-non set of systems, showing the average of the X_I point estimates for three metrics with $C = 0.42$ and $E = 0.01$. The final at-69,100 scores are now being overestimated when the judgment pool is shallow, but by less than the previous underestimates. Similarly, Table 1(d) shows the smoothed scores X_M with $C = 0.91$ and $E = 0.05$, for the same combinations of metrics and judgments. The smoothed estimates seem to have a more consistent trend of scores, especially from the 10,000 judgment starting point. The Interpolated and Smoothed predictors were not applied to the AP and NDCG metrics. Indeed, NDCG is relatively consistent in its value as the pool depth increases.

Separability Figure 4 shows system separability using P@10 as the number of judgments employed increases, where separability (sometimes also called discrimination) is the fraction of the possible system pairs that are identified as being statistically separable at the $p = 0.01$ confidence level. In this case, the fraction shown is relative to the $46 \times 45/2 = 1{,}035$ possible system pairs among the 46-non data set. The different curves within correspond to different score estimation method, C and E values used.

Surprisingly, it is the simplistic predictor X_S that generates the highest fraction of significant pairs at all depths of judgments compared to the other estimation methods when the underlying metric is P@10. Indeed, the high level of separability is attained despite the non-trivial residuals documented in Table 1(b) – the high separability is not just a matter of P@10 being a shallow metric and hence capable of being fully evaluated from a shallow pool. The same also holds true of the SDCG and RBP metrics – the highest separability arises with the simplistic predictor, and the Interpolative and Smoothed predictors give lower levels of separation between system pairs.

In behavior that is in agreement with the experimentation of other researchers (see, for example, Webber et al. [23]), AP tends to generate a greater fraction of separable system pairs than P@10, and NDCG is better again than AP. Figure 4 shows that this consistency happens irrespective of pool depth, and may be a consequence of the relative stability of the numeric values for the NDCG, as noted in Table 1(a). Similar outcomes are also noted by Webber et al. [23].

Reversals High separability rates are desirable, but only if the outcomes that are found to be significant are genuine ones. In particular, it is of concern if a metric asserts that some system significantly outperforms another when evaluated using a shallow pool, but the same conclusion cannot be reached when a more extensive set of relevance judgments is used. We call this situation a *reversal* – a system pair that is separable based on a prefix of the judgment set, but cannot be separated using the full set of 69,100 judgments (which is, of course, a prefix of the full all-documents all-topics judgment set). A system with a high separability rate might also suffer from a high reversal rate – in which case it is identifying spurious relativities between systems. Of course, the nature of significance testing itself allows some leeway in this regard – if 1,000 system pairs are evaluated, and 800 of them yield significance at the $p = 0.01$ level, then it would be unsurprising if a dozen of the system pairs did not yield significance on fresh topics and judgments.

Figure 5 shows the set of p values of the $46 \times 45/2 = 1{,}035$ system pairs making up the 46-non set, with the comparison based on use of the

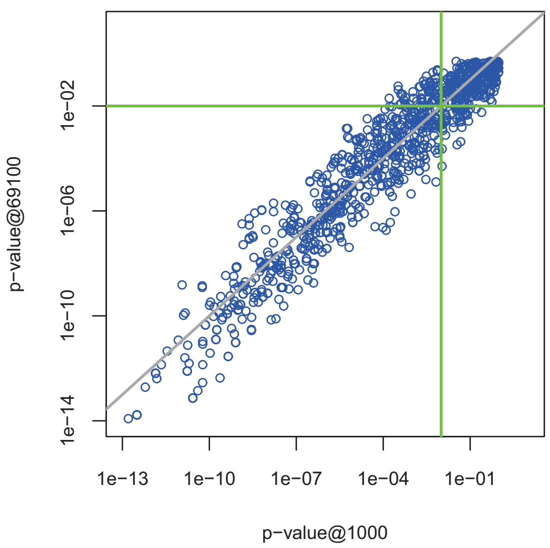


Figure 5: Using P@10 and X_S, with each point representing one system pair within the 46-non set, plotted according to the p-value computed over 50 topics, using a judgment pool of 1,000 outcomes (horizontal axis) and a judgment pool of all 69,100 outcomes (vertical axis).

t-test at the 0.01 confidence level. Each point plotted represents one system pair, with the horizontal location of the point determined by the p value computed for that pair when 1,000 judgments are being used (as previously, derived from the 59-con set over 50 topics), and the vertical location being determined by the p value that arises when the full 69,100 judgments are used.

Points in the lower-left quadrant of the graphs represent system pairs that are separable at the $p = 0.01$ level using both 1,000 and 69,100 judgments – that is, evaluations that are stable with respect to pool depth. The lower-right quadrant is also of interest – it indicates situations in which supplying more judgments improves separability, and points plotted in this zone can be regarded as being the payoff for performing deep judgments.

The quadrant of concern in Figure 5 is the upper-left one, which shows system pairs that were identified as being significantly different using shallow judgments, but for which that assessment was retracted once the full set of judgments was made available.

Table 2 draws all these ideas together, and lists separability percentages and reversal percentages (both as fractions of the $46 \times 45/2 = 1{,}035$ system pairs in the 46-non set of systems) for a wide range of metrics and point estimation mechanisms. The Simplistic prediction mechanism gives the greatest separability in each of the metrics, but also has a high rate of reversals. It thus appears that at least some of the separability advantage is illusory. On the other hand, the Interpolated predictor is less likely to yield a significant outcome at shallow pool depths, but compensates with a lower rate of reversed assessments. The two “deep” evaluation metrics, AP and NDCG, have the highest separability rates; but also have a relatively high rate of reversals.

Similar results were obtained for the metric P@10 when the same experiment was carried out using the TREC-8 Ad-Hoc Track data, consisting of 50 topics, 86,830 documents in the pool, and 71 runs contributed to the pooling (of a total 129 runs submitted).

5 Conclusion

All measurement involves uncertainty. When the measurement is of opinion-based outcomes, the uncertainties must be incorporated and managed; and when the cost of undertaking the measurement can be traded against repeatability and fidelity, the set of issues to be balanced becomes very large indeed. In this paper we have explored some of the consequences of shallow pooling in information retrieval experiments, and demonstrated that while simplistic predictions allow relatively high separability coefficients, there is also a higher rate of retraction of significance relationships as more judgments are performed. The Interpolative approach to predicting system scores is more robust in terms of reversals, but also less likely to find significance when the pool is shallow. It may be that these two facets of behavior are inevitable consequences of each other.

References

[1] K. Ali, C.-C. Chang, and Y. Juan. Exploring cost-effective approaches to human evaluation of search engine relevance. In *Proc. 32nd ACM SIGIR Conf.*, pages 802–803, Boston, USA, July 2009.

[2] C. Buckley and E. M. Voorhees. Evaluating evaluation measure stability. In *Proc. 23rd ACM SIGIR Conf.*, pages 33–40, Athens, Greece, July 2000.

[3] C. Buckley and E. M. Voorhees. Retrieval evaluation with incomplete information. In *Proc. 27th ACM SIGIR Conf.*, pages 25–32, Sheffield, England, July 2004.

[4] C. Buckley, D. Dimmick, I. Soboroff, and E. M. Voorhees. Bias and the limits of pooling. In *Proc. 29th ACM SIGIR Conf.*, pages 619–620, Seattle, WA, August 2006.

[5] B. Carterette, V. Pavlu, E. Kanoulas, J. A. Aslam, and J. Allan. Evaluation over thousands of queries. In *Proc. 31st ACM SIGIR Conf.*, pages 651–658, Singapore, July 2008.

[6] D. Hawking. Overview of the TREC-9 Web Track. In *Proc. 9th Text REtrieval Conf. (TREC-9)*, Gaithersburg, Maryland, November 2000.

[7] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[8] A. Moffat and J. Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems*, 27(1):1 – 27, 2008.

Metric	Predictor	Collection	Separability		Reversals
			1,000	69,100	
P@10	Simplistic	TREC-9	68.9%	67.5%	5.8%
P@10	Interpolated	TREC-9	58.6%	65.0%	2.2%
P@10	Smoothed	TREC-9	64.7%	62.7%	6.1%
SDCG	Simplistic	TREC-9	74.5%	74.9%	5.5%
SDCG	Interpolated	TREC-9	56.2%	71.6%	4.1%
SDCG	Smoothed	TREC-9	74.1%	67.7%	11.3%
RBP	Simplistic	TREC-9	74.1%	74.8%	4.9%
RBP	Interpolated	TREC-9	56.9%	72.9%	3.1%
RBP	Smoothed	TREC-9	72.1%	70.0%	7.8%
AP	Simplistic	TREC-9	69.1%	71.4%	5.4%
NDCG	Simplistic	TREC-9	75.8%	79.9%	4.3%

Table 2: Separability of metrics and predictors, and the fraction of reversals that arise when statistical testing based on a pool of 1,000 judgments is then extended to use all 69,100 judgements. In each case the evaluation is over all system pairs in the 46-non set of systems, with the percentages expressed as fractions of 1,035.

[9] A. Moffat, W. Webber, and J. Zobel. Strategic system comparisons via targeted relevance judgments. In *Proc. 30th ACM SIGIR Conf.*, pages 375–382, Amsterdam, July 2007.

[10] S. D. Ravana and A. Moffat. Score estimation, incomplete judgments, and significance testing in IR evaluation. In *Proc. AIRS Asia Information Retrieval Societies Conf.*, Taipei, Taiwan, December 2010. To appear.

[11] S. D. Ravana, L. A. F. Park, and A. Moffat. System scoring using partial prior information. In *Proc. 32nd ACM SIGIR Conf.*, pages 788–789, Boston, USA, July 2009.

[12] T. Sakai. Alternatives to Bpref. In *Proc. 30th ACM SIGIR Conf.*, pages 71–78, Amsterdam, July 2007.

[13] T. Sakai and N. Kando. On information retrieval metrics designed for evaluation with incomplete relevance assessments. *Information Retrieval*, 11(5):447–470, 2008.

[14] M. Sanderson and J. Zobel. Information retrieval system evaluation: Effort, sensitivity, and reliability. In *Proc. 28th ACM SIGIR Conf.*, pages 162–169, Salvador, Brazil, August 2005.

[15] K. Sparck Jones and C. J. Van Rijsbergen. Information retrieval test collections. *Journal of Documentation*, 32 (1):59–75, 1976.

[16] A. Trotman and D. Jenkinson. IR evaluation using multiple assessors per topic. In *Proc. 12th Australasian Document Computing Symp.*, pages 9–16, Melbourne, Australia, December 2007.

[17] E. M. Voorhees. The philosophy of information retrieval evaluation. In *Proc. 2nd Workshop of the Cross-Language Evaluation Forum (CLEF)*, pages 355–370, Darmstadt, Germany, September 2001.

[18] E. M. Voorhees. Topic set size redux. In *Proc. 32nd ACM SIGIR Conf.*, pages 806–807, Boston, USA, July 2009.

[19] E. M. Voorhees. Variations in relevance judgements and the measurements of retrieval effectiveness. In *Proc. 21st ACM SIGIR Conf.*, pages 315–323, Melbourne, Australia, August 1998.

[20] E. M. Voorhees and D. K. Harman. *TREC: Experiment and Evaluation in Information Retrieval*. The MIT Press, Cambridge, Mass., 2005.

[21] W. Webber and L. A. F. Park. Score adjustment for correction of pooling bias. In *Proc. 32nd ACM SIGIR Conf.*, pages 444–451, Boston, USA, July 2009.

[22] W. Webber, A. Moffat, J. Zobel, and T. Sakai. Precision-at-ten considered redundant. In *Proc. 31st ACM SIGIR Conf.*, pages 695–696, Singapore, July 2008.

[23] W. Webber, A. Moffat, and J. Zobel. The effect of pooling and evaluation depth on metric stability. In *Proc. 3rd EVIA Int. Work. Evaluating Information Access*, pages 7–15, Tokyo, Japan, June 2010.

[24] S. Wu and F. Crestani. Methods for ranking information retrieval systems without relevance judgements. In *Proc. ACM SAC Symp. on Applied Computing*, pages 811–816, Florida, USA, March 2003.

[25] J. Zhun, J. Wang, I. Cox, and V. Vinay. Topic (query) selection for IR evaluation. In *Proc. 32nd ACM SIGIR Conf.*, pages 802–803, Boston, USA, July 2009.

[26] J. Zobel. How reliable are the results of large-scale information retrieval experiments? In *Proc. 21st ACM SIGIR Conf.*, pages 307–314, Melbourne, Australia, August 1998.

Composition and Decomposition of Japanese Katakana and Kanji Morphemes for Decision Rule Induction from Patent Documents

Michiko Yasukawa and Hidetoshi Yokoo

Department of Computer Science
Gunma University
1-5-1 Tenjin-cho, Kiryu, Gunma, 376-8515 Japan

{*michi, yokoo*}*@cs.gunma-u.ac.jp*

Abstract *We propose a new method to construct a word list for rule induction from Japanese patent documents. For word segmentation in Japanese, statistical morphological analyzers have been used in many applications. However, the output of these morphological analyzers presents defects when analyzing unknown words, specifically words that contain Kanji/Katakana morphemes. Some words are overly segmented, and their original meanings are obscured. Furthermore, boundaries between compound nouns are uncertain, which impedes investigation in the initial stages of the application. In our method, we first perform morphological analysis to segment sentences into morphemes. Second, segmented compound words are filtered by character types and Katakana/Kanji morphemes in the compound words are concatenated. Third, the concatenated morphemes are truncated to reduce verbosity. Then, words comprising Katakana/Kanji are retained for use in a word list for rule induction. The experiment results show that our method is effective for extracting decision rules for patent classification.*

Keywords Information Retrieval, Natural Language Techniques and Documents

1 Introduction

Because of the growing demand for protection of intellectual property, patent documents have been increasing numerically on a global scale. To manage many documents, document classification is conducted using decision rules[1].

During the process of decision rule induction, a word list is referred to as a vocabulary of terms. To build the word list, word segmentation[2] is prerequisite. Although Japanese is an *unsegmented* language wherein word boundaries in texts are not clear, it is also a strongly *agglutinative* language, wherein boundaries between the morphemes (units smaller than words) are clear[3]. Therefore, Japanese texts are usually segmented into morphemes; these morphemes are used as terms in information retrieval systems in Japanese.

To segment Japanese sentences into morphemes, dictionary-based statistical morphological analyzers[4, 5] have been used in various applications. These Japanese morphological analyzers have high precision, and they are effective in many cases. However, morphemes suggested by a morphological analyzer can have numerous defects, especially when documents include many unknown Katakana/Kanji words. In general, patent documents are written using uncommon Katakana/Kanji jargon. Specifically, patent documents contain words or expressions of foreign origin; newly coined compound nouns representing novel technologies; names of uncommon substances, raw materials, medicines, or chemical products; etc. Therefore, morphological analyzers tend to produce incorrect results: they separate words excessively or wrongly. Table 1 presents some examples of excessive word segmentation by ChaSen[4], which is a commonly used morphological analyzer in Japanese. The left column shows keywords in patent documents. The right column shows results of word segmentation by the morphological analyzer. In the table, adjacent morphemes are separated by a colon (:) and displayed in the Key Word In Context (KWIC) format. The first three rows show examples that include the Kanji morpheme “空”(sora/kū). This particular Kanji symbolizes the word *sky* in English, but it can also have variant meanings such as *air*, *idle*, or *waste* depending on the context. The next three rows show examples that include the Kanji morpheme “導”(dō). This Kanji symbolizes the word *leading* in English. It also yields derivative meanings such as *assistance*, *conduction*, or *derived* depending on the adjacent Kanji characters. The last four rows present examples that include the Katakana “レ”(re). This Katakana character transliterates the syllable “le” or “re” in foreign words. To convey a meaning, the sequence of Katakana characters in each keyword should not be separated. If the Katakana sequences are decomposed into fragmented morphemes of Katakana sequences, then identifying documents by keywords would be difficult. For example, a user who wants to search

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

Table 1: Examples of excessive word segmentation

	Example keywords	Word segmentation by a morphological analyzer
1.	水陸空	**水陸**: **空** :兼用:輸送:機
	suirikukū	multi-use transport *on land, at sea, and in the air*
2.	空運転	ポンプ: **空** :**運転**:防止:用:強制:停止:信号
	karaunten	a forced stop signal for prevention of *idle running* of pumps
3.	空缶	リサイクル:用: **空** :**缶**:箱
	akikan	a box for *waste can* recycling
4.	聴導犬	**聴**: **導** :**犬**:用:警報:音:発生:回路
	chōdōken	an alarm-tone-generating circuit for *hearing assistance dog*
5.	骨導	**骨**: **導** :ヘッド:セット
	kotsudō	a *bone-conduction* headset
6.	導関数	二:次: **導** :**関数**
	dōkansū	a second order *derivative*
7.	ソレノイド	電磁:**ソ**: **レ** : **ノイ**:**ド**
	sorenoido	an electromagnetic *solenoid*
8.	レジン	義歯:用: **レ** :**ジン**
	rejin	*resin* for artificial teeth
9.	レコーダ	カセットテープ: **レ** :**コーダ**
	rekōda	a cassette tape *recorder*
10.	スフレ	**スフ**: **レ** :生地
	sufure	a mixture of ingredients for baking *souffle*

for “solenoid”[1] might also receive documents about “souffle”[2] because they have the common Katakana morpheme “レ” in “ソ:レ:ノイ:ド”(sorenoido) for *solenoid* and “スフ:レ”(sufure) for *souffle*.

When a morphological analyzer separates a long sequence of compound words into words, the situation becomes more complicated. The boundaries between compound words can be ambiguous. Therefore, word segmentation using only a statistical morphological analyzer is insufficient. Numerous combinations of shorter compound words can exist in a lengthy compound word. To select optimal words for the target application, character types and statistics might be used to determine the plausible word boundaries. In addition, a word list for the rule induction should contain multiple choices that can be informative for additional processes to meet the final objective of the application.

In Section 2, we describe our objective, i.e., to induce decision rules from patent documents. Furthermore, we address the problem of incorrectly segmented morphemes. In Section 3, we describe our proposed method to compose and decompose the segmented morphemes. In Section 4, we describe experiments that show that our method can be effective for improving the quality of the rules induced from patent documents.

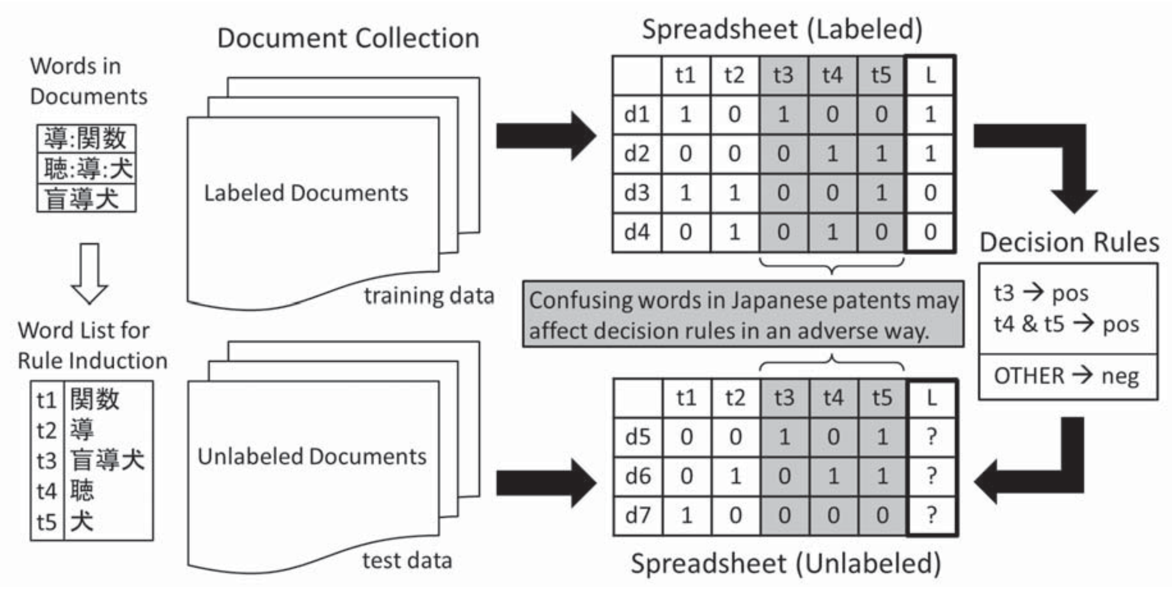
2 Decision Rules for Patent Classification

As described in this paper, our goal is to improve the performance of patent classification using decision rules. The objective of the decision rules is to acquire appropriate labels for newly arrived documents. The rules can also suggest useful keywords used to search in documents. The rules comprise words, which have meaning. Therefore, they can suggest reasons for reaching a conclusion. Decision rules can be more predictive and insightful than categorizers based on scores or measures of similarity[1].

A schematic of rule induction from documents is presented in Figure 1. The objective of decision rules is to distinguish one class from the other. Consequently, prediction is conducted using binary classification wherein the positive (interesting) documents are separated from the negative (not interesting) ones. In Figure 1, both labeled documents and unlabeled documents are transformed into a spreadsheet because the values in a spreadsheet are easier to handle than in an unstructured document. In the spreadsheets, rows represent documents and columns represent words, except the column on the extreme right that shows the labels for the documents. The values in the columns for words are one or zero, respectively indicating the presence or absence of each word in the documents. The values of labels are also one or zero, respectively denoting a positive document or a negative document. Once the spreadsheet of labeled documents is obtained, the decision rules can be induced. The induced rules

[1] a current-carrying coil of wire
cf. http://www.thefreedictionary.com/solenoid

[2] light fluffy dish of eggs
cf. http://www.thefreedictionary.com/souffle



	t1	t2	t3	t4	t5	L
d1	1	0	1	0	0	1
d2	0	0	0	1	1	1
d3	1	1	0	0	1	0
d4	0	1	0	1	0	0

	t1	t2	t3	t4	t5	L
d5	0	0	1	0	1	?
d6	0	1	0	1	1	?
d7	1	0	0	0	0	?

Figure 1: Rule Induction from Japanese Patent Documents

are applied to the unlabeled documents to predict the labels for those documents.

The primary steps in rule induction are the following. (1) Find a set of rules to separate the two classes, i.e. positive vs. negative. (2) Iteratively prune the rule set into simpler rule sets. (3) Select the best set of decision rules that are fairly simple and produce fewer errors.

Using the induced decision rules, document categorization with high precision is achievable for documents written in English[1]. However, when they are applied to Japanese patent documents, the categorization precision declines. This is true because word segmentation in Japanese is not successful, especially in patent documents. Japanese includes multiple intermingled writing systems for which the morphemes are not separated in written text. To obtain morphemes in sentences, morphological analysis is generally conducted.

During rule induction, the columns of the spreadsheet in Figure 1 do not represent mere words, but are presumed to serve as important keywords that represent concepts in the documents. Because dictionary-based statistical morphological analyzers refer to their own dictionaries and because dictionaries do not include all the words in Japanese, some words are excessively or wrongly separated, as shown in Table 1. When the word "盲導犬"(mōdōken; guide dog) is analyzed as a single word, native speakers of Japanese would expect that "聴導犬"(chōdōken; hearing assistance dog) is also analyzed as a single word because they are comparable types of the word *assistance dog*, which is represented by the Kanji sequence "導犬"(dōken). However, morphological analyzers separate the second one into three Kanji morphemes "聴"(chō) for *hearing*, "導"(dō) for *assistance*, and "犬"(ken) for *dog* because it is not included in their dictionaries. The first is not separated because it is specified as a single noun word in their dictionaries. In another example, morphological analyzers incorrectly separate the unknown word "スフレ"(sufure) into two morphemes "スフ"(sufu) and "レ"(re). Here, "スフ"(sufu) in Japanese is an abbreviation for "ステープルファイバー"(sutēpuru faibā), which means *staple fiber* in English, and and "レ"(re) is *re*, which is the second tone of the diatonic scale in solfeggio. When Japanese morphological analyzers process Japanese patent documents, such mistakes occur frequently. Therefore, it is necessary to amend the output of morphological analyzers when constructing a word list from patent documents.

3 Composition and Decomposition of Morphemes

In this section, a method for constructing a word list from Japanese patent documents is described. The method consists of two processes: (1) composition of Katakana/Kanji morphemes and (2) decomposition and re-composition of morphemes. In the following sections, these processes are explained in detail.

3.1 Composition of Katakana/Kanji Morphemes

In patent documents, Katakana/Kanji words are used widely to describe novel concepts in science and technology. Katakana are Japanese characters that are mainly used for spelling loan words. Many words used in terminology of science and technology are transliterated from foreign characters into Katakana according to their original pronunciation and spelling. Kanji are ideographic representations of objects and ideas. In general, nouns are written in Kanji, although verbs and adjectives are written using a combination of Kanji and Hiragana[2]. Although some nouns are

Table 2: Examples of Long Compound Words

1.	(a) を持たせたことを特徴とする油圧減速機常備小型自走式クレーン搭載小型杭打機に (b) 油圧:減速:機:常備:小型:自:走:式:クレーン:搭載:小型:杭:打:機 (c) yuatsu gensoku ki jōbi kogata ji sō shiki kurēn tōsai kogata kui uchi ki
2.	(a) である、高域劣化補償ガードインターバル挿入式直交周波数分割多重変調装置と (b) 高域:劣化:補償:ガード:インターバル:挿入:式:直交:周波数:分割:多重:変調:装置 (c) kōiki rekka hoshō gādo intābaru sōnyū shiki chokkō shūhasū bunkatsu tajū henchō sōchi

(a): compound words in context, (b): extracted morphemes, (c): reading of the compound words

written in Hiragana or Katakana as well as Kanji, adults prefer Kanji to Hiragana or Katakana to write formal documents. Patent documents are written in a formal tone by adults. Consequently, Kanji are in heavy usage.

Although e-mail messages contain informal spelling alternatives[7] and web texts contain slang words[6], patent documents are written in a distinctive style. Therefore, slang words do not present important issues in patent documents. However, extremely lengthy compound words are frequently used in patent documents. They are difficult to handle during word segmentation by morphological analyzers because these words are not learned from training data. These compound words are coined frequently by the authors of patent documents to describe novel technologies. Table 2 shows some difficult compound words. The first one includes 23 characters. The second one, in fact, includes 31 characters.

In the composition process, we first let morphological analyzers break sentences into morphemes. Generally speaking, morphological analyzers try to carry out word segmentation to the greatest possible extent when they encounter unknown words. Then, we identify certain morphemes according to the types of characters they include. In particular, the following morphemes are identified and extracted.

- A morpheme that is made up only of Kanji.
- A morpheme that begins only with Katakana and for which the latter part, if any, is made up of Katakana or circumflexes.

Finally, we extract Katakana/Kanji morphemes and compound them with an interposing colon (:) between adjacent morphemes, as shown in the (b) parts in Table 2. All Hiragana, Latin alphabet characters, numbers, and punctuation marks are filtered out during this process.

3.2 Decomposition and Re-composition of Morphemes

In the previous process, a sequence of morphemes is composed to be included in the word list for rule induction. Although many of the composed morphemes are very good restorations of original compound words, some inlcude attached insignificant morphemes in front or behind them.

In patent documents, most important morphemes tend to be located in the middle of a word; ancillary morphemes are adhered to them. Some morphemes are added only as a matter of form, or as stereotypical expressions in patent documents. Others are attached to rephrase the word, or to broaden the extent of the patent. Japanese is an agglutinative language. Therefore, additional morphemes are adherent to the words, apparently forming parts of the compound words. However, if a morpheme is attached to any word repeatedly in the same patent, then it is presumably a trivial affix. Some morphemes might be repeated in the same category of patents, or in the whole collection of patent documents. The most typical affixes in Japanese patent documents are morphemes representing "上記" for *aforementioned*, "等" for *such as*, "装置" for *apparatus*, or "手段" for *means*. Although some affixes have a significant number of document frequencies, others are not always outstanding. A repetitive morpheme that is meaningless in many documents can be a part of important compound words in some documents. Therefore, it is necessary to extract plausible important keywords in the pre-process and let the main process decide which one to choose.

To truncate lengthy compound words, the first and the last morphemes should be removed from the composed morphemes. The composed morphemes are decomposed and re-composed as shown in Figure 2. When a composed morpheme includes only two morphemes, it is not truncated. When a composed morpheme includes three morphemes, the first morpheme is truncated and the remainder of morphemes are re-composed. Successively, the last one is truncated and the remaining morphemes are re-composed. Then, two re-composed morphemes are obtained. When a composed morpheme includes four or more morphemes, the truncation is performed, respectively, at the first, the last, and the first and the last morphemes. Then, three re-composed morphemes are obtained. Finally, the truncated morphemes, namely, re-composed morphemes, are added to the word list as well as the original ones. The procedure of our method, including decomposition and re-composition of morphemes, is described in the following algorithm.

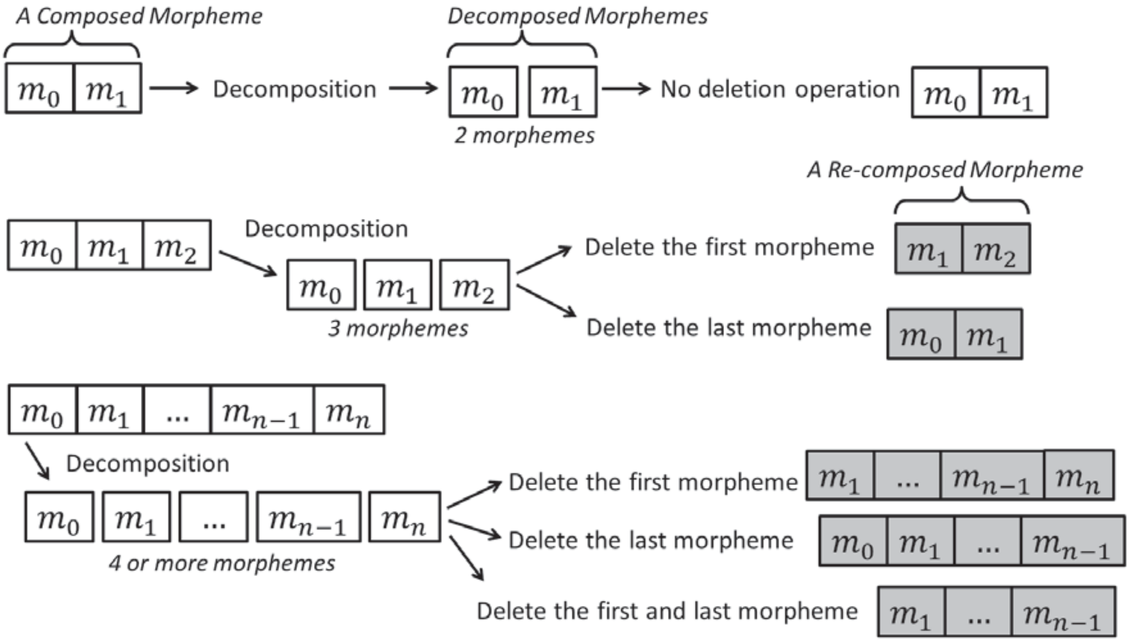


Figure 2: Composed, Decomposed and Re-composed Morphemes

Algorithm— Word List Construction

```
Initialize: $L$ = word list;
for every morpheme $M_i$ in documents do
  if $M_i$ is noun, verb, or adjective then
    add $M_i$ to $L$
  endif
endfor
for every composed morpheme $C_j$ do
  add $C_j$ to $L$
  if $C_j$ has three or more morphemes then
    decompose and re-compose morphemes in $C_j$
      without the first morpheme and
      add the re-composed morpheme to $L$
    decompose and re-compose morphemes in $C_j$
      without the last morpheme and
      add the re-composed morpheme to $L$
  endif
  if $C_j$ has four or more morphemes then
    decompose and re-compose morphemes in $C_j$
      without the first and the last morphemes and
      add the re-composed morpheme to $L$
  endif
endfor
```

4 Experiment

4.1 Patent Classification by Decision Rules

An implementation of the text categorization method proposed in [1], which is a software tool kit called RIKTEXT[11], is available. We used this software to evaluate our proposed method in a patent classification application. The classifier performance is assessed using three ratios: precision, recall, and the F-measure[12]. Precision is the ratio of the number of correct positive predictions to the number of positive predictions. Recall is the number of correct positive predictions to the number of positive class documents. The F-measure F is derived using the following equation.

$$F = \frac{2}{1/P + 1/R} \tag{1}$$

where P is the precision and R denotes the recall.

4.2 Datasets for Experiments

For the experiment, we must prepare document collection that consists of training data and test data, as shown in Figure 1. Both training and test data must be labeled respectively with a one or zero, indicating a positive document or a negative document. In the past NTCIR Patent Retrieval Task, the Classification Subtask was conducted. A test collection for patent classification was released in the subtask. This test collection is well designed, but it does not meet the requirements for our experiment. In the test collection, the system must determine one or more patent categories for each patent document. In our experiment, the system must determine one or zero for the label of each patent document in the test data. Therefore, we construct the test collection that satisfies the experimental requirements.

For the experiment, we use a document collection constructed in NTCIR-6 Patent Retrieval Task[8]. The document collection consists of 3,496,352 Japanese patent applications published during 1993–2002. The number of search topics is 2,908. Each document is given one or more International Patent Classification (IPC) codes[9]. We used these IPC codes to assign labels for positive/negative documents. For each session of classification, the positive documents have the same IPC codes although the negative documents have different IPC codes from positive ones.

The IPC codes consist of five level layers, but the lowest level is too specific and the upper level is too general. Therefore, we focused on the middle level layer: the 3rd level. For example, a document that is given the IPC code "A01M 21/00" is a positive document for the 3rd level IPC code "A01M." Any documents that are given different IPC codes from "A01M" are negative documents in this case. More specifically, if a document attached "A01M 21/00," which represents "Apparatus for destruction of unwanted vegetation, e.g. weeds," then this document is a positive document for the category "A01M," which represents "Catching, trapping or scaring of animals. Apparatus for the destruction of noxious animals or noxious plants." A document attached "A01H 3/00," which represents "Processes for modifying phenotypes," is a negative document.

Regarding pre-processing, first, the IPC codes were extracted from all documents. Then, documents were analyzed using the morphological analyzer ChaSen[4] to extract morphemes. After pre-processing, patent documents that satisfy the following conditions were collected.

- Only one IPC code is given: no multiple IPC codes are given for each document in the experimental document collection.
- The 5th level of the given IPC code is '00' that means the main group of each category.
- The number of morphemes in the document is not extreme. We used documents that contain 100 or more, and 10000 or fewer morphemes.

In this way, 148,892 documents were collected. From the collected documents, datasets of two types were produced: positive/negative sets per IPC (dataset-1) and positive/negative sets per search topic (dataset-2).

For dataset-1, IPC codes that include a moderate number of documents were selected. Specifically, we selected IPC codes that include 500 or more documents and 1000 or fewer documents. In this way, 51 IPC codes and 60,554 documents were aggregated. For every IPC code, 300 positive and 1,200 negative documents were randomly picked out from the aggregated documents. Then, 200 positive and 800 negative documents were used as training data. Furthermore, 100 positive and 400 negative documents were used as test data.

For dataset-2, document search was conducted for every search topic. For document search, GETA[10] was used. For every document search, the search query consisted of nouns, verbs, and adjectives in the claim part of a search topic. As for term weighting in the document search, the following pivoted normalization of TF-IDF weight, which is proposed in [13], was used.

$$w_{d,t} = \frac{1+\log(f_{d,t})}{1+\log(avef_d)} \times \frac{(1+\log(f_{q,t}))\times idf_t}{avedlb + S\times(dlb_d - avedlb)} \quad (2)$$

In that equation, d represents a unique document, t represents a unique term, $f_{x,t}$ is the frequency of term t in x, $idf_t = 1 + \log(N/n_t)$ is the inverse document frequency of term t, $avef_x$ denotes the average frequency of each term in x, dlb_x is the number of unique terms in x, $avedlb$ represents the average of dlb_x in the collection, and $S = 0.2$ is a constant.

From every search result, 500 positive and 1,000 negative documents were obtained. Then, the top 200 positive and top 300 negative documents are used as test data. The next 300 positive and the next 700 negative documents are used as training data. Search topics that have fewer than 500 positive documents were discarded. In this way, 1,129 search topics and 137,752 documents were aggregated.

All labeled documents for dataset-2 were obtained through document searching. Although no IPC codes are common between positive and negative classes, both positive and negative documents are controlled to become similar in dataset-2. For this reason, rule induction from dataset-2 is expected to be difficult.

4.3 Experimental Results

For comparison, components of the word list for rule induction were varied. Specifically, for components of the word list, we used (1) only nouns; (2) nouns, verbs, and adjectives; (3) nouns, verbs, adjectives, and compound morphemes; (4) only compound morphemes; (5) compound and re-compound morphemes; (6) nouns, verbs, adjectives, compound morphemes, and re-compound morphemes. The last one is our proposed method. For every condition and dataset, experimental patent classification was performed. Table 3 and Table 4 respectively present the classification performance in dataset-1 and dataset-2. In the tables, Prec and Rec respectively represent precision and recall. As shown in the tables, the proposed method has the highest average F-measure in both dataset-1 and dataset-2. The number of induced decision rules is shown in Figure 3. When the word list consists of compound morphemes, many decision

Table 3: Patent Classification (dataset-1)

component of	min.			max.			avg. per IPC code		
word list	Prec	Rec	F-measure	Prec	Rec	F-measure	Prec	Rec	F-measure
(1) n	72.53	36.00	52.94	100.0	95.00	97.44	91.50	78.33	83.84
(2) n+va	73.12	36.00	52.94	100.0	96.00	97.44	91.61	78.75	84.10
(3) n+va+comp	71.84	36.00	52.94	100.0	96.00	97.44	91.87	78.73	84.17
(4) comp	60.00	2.00	3.92	100.0	80.00	87.91	92.12	50.49	63.30
(5) comp+re	64.71	8.00	14.29	100.0	90.00	93.62	91.95	59.12	70.60
(6) n+va+comp+re	73.68	36.00	52.94	100.0	96.00	97.44	91.50	79.29	**84.41**

n: nouns, va: verbs and adjectives, comp: compound morphemes, re: re-compound morphemes

Table 4: Patent Classification (dataset-2)

component of	min.			max.			avg. per search topic		
word list	Prec	Rec	F-measure	Prec	Rec	F-measure	Prec	Rec	F-measure
(1) n	54.90	19.00	29.92	100.0	100.0	99.75	85.52	73.60	78.20
(2) n+va	54.72	19.00	30.52	100.0	100.0	99.75	85.90	73.92	78.54
(3) n+va+comp	61.54	24.00	35.96	100.0	100.0	99.75	86.49	74.39	79.12
(4) comp	60.19	2.50	4.83	100.0	99.50	99.50	87.14	58.61	68.46
(5) comp+re	59.84	18.00	28.91	100.0	99.50	98.23	87.37	63.52	72.25
(6) n+va+comp+re	61.41	14.00	23.43	100.0	100.0	99.75	86.82	74.37	**79.26**

n: nouns, va: verbs and adjectives, comp: compound morphemes, re: re-compound morphemes

rules are generated because compound morphemes are conjunction of words and they are more specific than single morphemes. For this reason, without single morphemes ((4), (5) in Table 3, 4), patent classification tends to produce low recall. However, without considering compound morphemes ((1), (2) in Table 3, 4), patent classification tends to produce low precision. The number of extracted keywords is presented in Figure 4. As shown in the figure, our proposed method can retain the largest size of vocabulary to the classifier because it makes the best of components of all types. Therefore, the proposed method is considered to be optimal when extracting decision rules from Japanese patent documents.

5 Related Work

Although most European languages are space-delimited languages, Asian languages such as Chinese and Japanese are unsegmented languages[3]. In both unsegmented and space-delimited languages, specific challenges are posed by word segmentation. In Chinese, word segmentation methods based on Conditional Random Field (CRF) have been proposed[14, 15]. In German, a method for splitting compound words using a Support Vector Machine (SVM) has been proposed[16]. In general, such dictionary-based statistical methods are effective. However, exceptional compound words are not analyzed correctly by those methods because these words are not learned from training data. Regarding Japanese studies, some previous works have described extraction of unknown words: methods particularly addressing Kanji[17, 18], methods particularly addressing Katakana[19, 20], and a method particularly addressing morphological aspects[6]. Our method particularly addresses both Kanji and Katakana, and investigates character types rather than morphological aspects of compound words.

6 Conclusion

We proposed a method to construct a word list for rule induction from patent documents. Patent documents include unusual lengths of compound words. Consequently, morphological analyzers tend to produce incorrect results during word segmentation processing. When excessive word segmentation is performed on Katakana/Kanji words, their original meanings are obscured. They are adversely affected by such segmentation in applications such as information retrieval and text categorization.

Our method specifically addresses Katakana/Kanji that are used widely in Japanese patent documents. First, word segmentation is performed using a morphological analyzer. The resultant morphemes are examined using character types. Second, Katakana/Kanji morphemes are identified, extracted, and composed. Third, the composed morphemes are decomposed and re-composed to remove ancillary

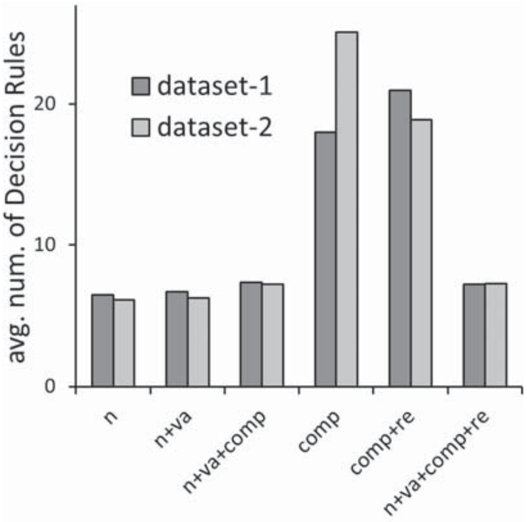


Figure 3: Induced Decision Rules

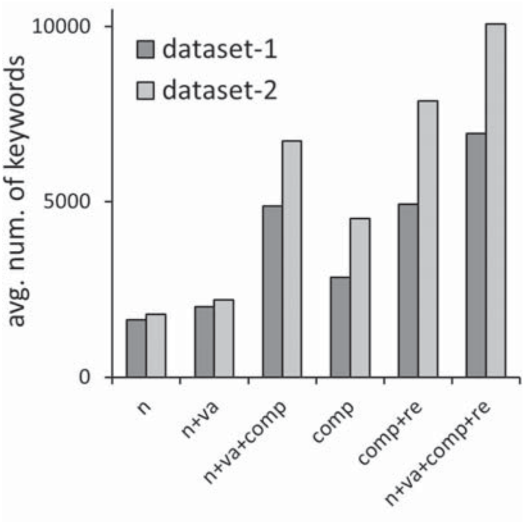


Figure 4: Extracted Keywords

morphemes. In the evaluation experiment, we applied our method to patent classification using decision rules. The re-composed morphemes and the original composed morphemes are added to the word list for rule induction to increase the number of extracted keywords. Experimental results show that our method increases the text categorization precision.

Nevertheless, there is room for additional truncation and normalization for extremely lengthy compound words. To cope with this problem, we aim to truncate composed morphemes iteratively in an efficient way. We are also planning to produce a stop word list to facilitate construction of a word list.

Acknowledgements We are grateful to the NTCIR project for providing NTCIR-6 patent collections.

References

[1] Apté, C., Damerau, F., Weiss, S.M.: Automated Learning of Decision Rules for Text Categorization. ACM Trans. Inf. Syst. 12(3): 233–251 (1994)

[2] Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press. (2008)

[3] Dale, R., Moisl, H., Somers, H.: Handbook of Natural Language Processing. CRC Press. (2000)

[4] ChaSen: http://chasen-legacy.sourceforge.jp/

[5] Mecab: http://mecab.sourceforge.net/

[6] Murawaki, Y., Kurohashi, S.: Online acquisition of Japanese unknown morphemes using morphological constraints. EMNLP 08, pp. 429–437 (2008)

[7] Nishimura, Y.: Linguistic Innovations and Interactional Features of Casual Online Communication in Japanese. JCMC, vol.9, no.1 (2003)

[8] Fujii, A., Iwayama, M., Kando, N.: Overview of the Patent Retrieval Task at the NTCIR-6. Proc. NTCIR-6 Workshop Meeting, pp. 359–365 (2007)

[9] WIPO: International Patent Classication (IPC). http://www.wipo.int/classifications/ipc/en/

[10] Generic Engine for Transposable Association (GETA). http://geta.ex.nii.ac.jp/e/

[11] Indurkhya, N.: RIKTEXT: Rule Induction Kit for Text. http://www.data-miner.com/riktext.pdf (2004)

[12] Weiss, S.M., Indurkhya, N., Zhang, T., Damerau, F.J.: Text Mining Predictive Methods for Analyzing Unstructured Information. Springer. (2005)

[13] Singhal, A., Buckley, C., Mitra, M.: Pivoted Document Length Normalization. SIGIR 96, pp. 21–29 (1996)

[14] Tseng, H., Chang, P., Galen, A., Jurafsky, D., Manning, C.: A Conditional Random Field Word Segmenter. 4th SIGHAN Workshop on Chinese Language Processing. pp. 168–171 (2005)

[15] Fuchun, P., Feng, F., McCallum, A.: Chinese segmentation and new word detection using conditional random fields, COLING 04. pp. 562–568 (2004)

[16] Alfonseca, E., Bilac, S., Pharies, S.: German Decompounding in a Difficult Corpus, CICLing 08, pp. 128–139 (2008)

[17] Watanabe, Y., Murata, M., Takeuchi, M., Nagao, M.: Document Classification Using Domain Specific Kanji Characters Extracted by X2 Method. COLING 96, pp. 794–799 (1996)

[18] Ando, R. K. Lee, L.: Mostly-Unsupervised Statistical Segmentation of Japanese Kanji Sequences. Natural Language Engineering 9 (2): 127–149 (2002)

[19] Seki, K., Hattori, H., Uehara, K.: Generating diverse katakana variants based on phonemic mapping. SIGIR 08, pp. 793-794 (2008)

[20] Nakazawa, T., Kawahara, D., Kurohashi, S.: Automatic Acquisition of Basic Katakana Lexicon from a Given Corpus. IJCNLP 05, pp. 682–693 (2005)

Evaluating the Effectiveness of Visual Summaries for Web Search

Hilal Al Maqbali *Falk Scholer* *James A. Thom* *Mingfang Wu*
School of Computer Science and Information Technology
RMIT University
GPO Box 2476, Melbourne 3001
Victoria, Australia

{*hilal.almaqbali,falk.scholer,james.thom,mingfang.wu*}@*rmit.edu.au*

Abstract

With ever-increasing amounts of information on the World Wide Web, an effective interface for displaying search results is required. Recent studies have developed various novel approaches for visual summaries, aiming to improve the effectiveness of search results. In this study we evaluate the effectiveness of four types of visual summary: thumbnails, salient images, visual snippets and visual tags. Fifty participants carried out five informational topics using five different interfaces. The results show that visual summaries significantly impact on the behavior of users, but not on their performance when predicting the relevance of answer resources. Users spend significantly less time looking at the textual components of summaries with the visual summary interfaces. Comparing the performance of users in predicting the relevance of answer pages with a text interface versus visual interfaces suggests that the tested visual summaries can mislead users to select non relevant items on informational search topics.

Keywords Information Retrieval, User Studies Involving Documents, Web Documents, Visual Summaries, Eye Tracking.

1 Introduction

The amount of information on the World Wide Web has been increasing exponentially; and search engines are the key tool for supporting users in finding information. Search results presentation and organisation are important components that impact on the overall search effectiveness [1, 2]. Existing search engines not only show textual summaries (such as a page title, a short textual snippet, and a URL), but also provide visual features. One of the most common types of feature is the visual summary, such as a thumbnail or a dominant image [14].

Although popular search engines such as Google historically focused more on improving textual summaries for each result page, they have started showing visual summaries for some of the top search results. Other search engines, such as Middlespot, Nexplore and Viewzi, display a visual summary for every result in their answer list [1].

It has been said that one image is worth a thousand words. Humans can digest the meaning of an image quicker than text; in the time that a user spends to understand the gist of an image, a user can read only one to four words [7]. Visual summaries have been shown to significantly help users in the judgment of web search results in many studies [9, 14, 15, 19, 21, 22]. Visual summaries are helpful for refinding previously visited web pages and can also provide hints for users when the search tasks are confusing or when users are not proficient at them.

In this study, we investigate the impact of different approaches of visual summaries for informational tasks on user behavior and analyse the time spent looking at each component of the search results presentation (title, text snippet, URL and visual summary). We compare visual snippets, visual tags, excerpt images, and thumbnails. Each one of these visual summaries is presented on an interface together with a text summary. We investigate the following questions:

1. Does providing additional visual summaries with search results improve the ability of users to predict the relevance of search results?
2. How do visual summaries impact on user behavior, particularly on text summaries when additional visual summaries are presented?
3. How do users interact with different components of a results screen, for those search results that include visual summaries?
4. How does the presence of visual summaries affect task completion time?

Our analysis shows that users spend significantly less time looking at textual summaries when visual summaries were available. However, overall, the results suggest that visual summaries do little to increase user performance with informational topics.

This paper is organised as follows: in Section 2, related work is reviewed. In Section 3, we describe our experiment on design including the visual summaries, users and topics. Experimental results are analysed in

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

Section 4, and discussion and conclusion are presented in Section 5.

2 Related work

Dziadosz [9] described the interaction between user and information retrieval systems, and summarized these into three steps: query formulation, relevance prediction, and relevance evaluation. Visual summaries can help to improve the relevance prediction as shown in many studies [9, 14, 15, 19, 21, 22].

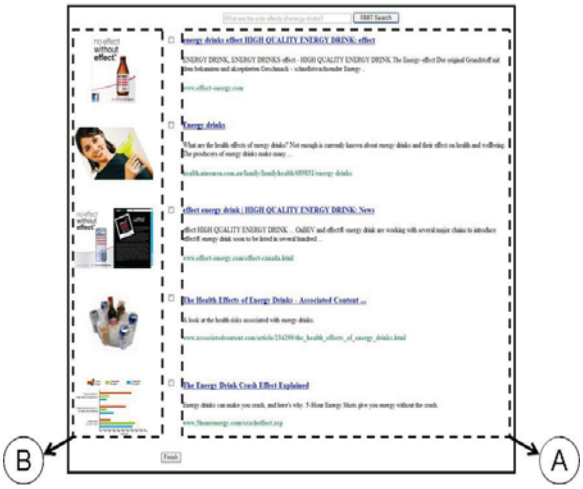
Several novel approaches have been developed for visual summaries to improve user performance in finding desired information. Some of these approaches use the snapshot of a web page, such as a thumbnail or an enhanced thumbnail [21], whereas others use a salient picture within the retrieved web page such as a salient image [14] or visual snippet [19].

A thumbnail is a miniature image of a web page. This is the most common type of visual summary and has been evaluated in many studies [3, 6, 8–10, 12, 16, 20–22]. Thumbnails help users to recognise the layout of the retrieved web page. They not only make it easy to recognise the web page if it has been seen before, but can also provide users with relevant visual hints for their queries in the form of a picture, table or website logo.

Woodruff et al. [21] develop a novel approach for a visual summary called **an enhanced thumbnail**, which highlights and enlarges the query terms within the thumbnail images. They compared enhanced thumbnails with plain thumbnails and with text summaries. Four different search tasks were used where users were asked to find a picture, homepage, e-commerce website, or side-effects of a given drug (informational query). The results show that visual summaries reduced the number of visited pages to find the answer for a given search task. Also, they found no significant difference on performance and time spent to answer the informational queries.

Teevan et al. [19] developed a **visual snippet** which combines the page title, a salient image and a logo of the website. Twelve topics were used in the study, four for each type (homepage, shopping, medical information). Analysis shows that visual snippets are more useful than plain thumbnails, particularly for revisitation (refinding a previously seen web page). The study also discussed the possibility of generating visual snippets automatically.

Li et al. [14] examined the effectiveness of presenting **excerpt (salient) images** with search results, by examining two interfaces, one with text summaries only and the other with both text summaries and excerpt images. Two types of queries were used in the study, informational and navigational. The results showed that excerpt images are helpful, and can be generated for almost all query types. For example, according to the experimental results, excerpt images decreased the time spent by the user on informational queries by 30.4% compared with the time spent on text summaries without excerpt images.


What are the side effects of energy drinks? RMIT Search

□ energy drinks effect HIGH QUALITY ENERGY DRINK: effect
ENERGY DRINK, ENERGY DRINKS effect - HIGH QUALITY ENERGY DRINK The Energy-effect Der original Grundstoff mit dem bekannten und akzeptierten Geschmack - schnellwirkender Energy ...
www.effect-energy.com

□ Energy drinks
What are the health effects of energy drinks? Not enough is currently known about energy drinks and their effect on health and wellbeing. The producers of energy drinks make many ...
health.ninemsn.com.au/family/familyhealth/689831/energy-drinks

□ effect energy drink | HIGH QUALITY ENERGY DRINK: News
effect HIGH QUALITY ENERGY DRINK ... OzBEV and effect® energy drink are working with several major chains to introduce effect® energy drink soon to be listed in several hundred ...
www.effect-energy.com/effect-canada.html

□ The Health Effects of Energy Drinks - Associated Content ...
A look at the health risks associated with energy drinks.
www.associatedcontent.com/article/234299/the_health_effects_of_energy_drinks.html

□ The Energy Drink Crash Effect Explained
Energy drinks can make you crash, and here's why. 5-Hour Energy Shots give you energy without the crash.
www.5hourenergy.com/crasheffect.asp

Finish

Figure 1: Salient image interface: (A) Text summary region. (B) Visual summary region.

A study by Jiao et al. [11] was conducted to evaluate four types of visual summaries: internal image (the dominant image in a web page); external image (a representative image from an external page); visual snippets; and, thumbnails. The study evaluated these visual summaries in two phases. In the first phase, participants were given the visual summaries and asked to type descriptions about the expected content of the related web page. Several hours later, in the second phase, researchers investigated how the visual summaries affected the recall of the web pages visited in phase 1. The results show different types of visual summary work better on particular types of web pages and search tasks.

Tag clouds have become a popular method for visualising information on the web. Tag clouds visualise the most frequently used words in a document by showing the relative importance of terms using different font sizes, weight or color. Many studies [4,17,18] show that tag clouds can provide effective cues about the content of text search results for users. We developed a novel approach (called visual tags) for the visual summary which combines the tag clouds of a document and its snapshot. Our hypothesis is that these combined visual tags will provide useful hints for users about the content of the retrieved web page.

3 Experimental methodology

In order to evaluate the effectiveness of different approaches for visual summaries, and study the impact of these visual summaries on user seeking behavior and performance, we conducted a user study that involved a series of five informational search topics using different search interfaces where visual summaries are a primary component of the search results presentation.

3.1 Experimental setup

In our user study, the participants were mostly undergraduate and high school students with some interest in computer science visiting RMIT University at the 2010 Open Day. A plain language statement was given to the subjects to outline the purpose of the experiment, the procedure, the tasks to be performed, and the data to be collected. Based on this information, 65 participants chose to take part in the experiment. However, due to interruptions and difficulty with calibrating the eye tracking for some volunteers (we eliminated users with less than 80% capture accuracy), the collected data of only 50 participants is included in the analysis. A short oral presentation about the visual summaries was given to each participant, but no training was given on the interfaces to be used.

Each participant was asked to evaluate items in search result lists for five informational search topics, each with a different interface. For each topic, five answer items were shown on a single page. Participants used the mouse to select all items that they considered to be relevant to the given topic. Participants were not able to browse the actual web pages embedded in the text search results, relying solely on the search result page given. Note that users were presented with a fixed search results list for the topic, and did not engage in interactive searching.

Data was collected using a Tobii T60 eye tracker. This non-invasive device calculates the exact point of a user's gaze using a geometrical model. Since all the search results were presented on a single screen page, participants did not spend extra time or visual attention having to scroll the search results page.

3.2 Interfaces

Five interfaces were designed for this experiment, each presenting exactly the same text summary, but with different visual browse features (visual summaries), except for the text interface which presents only text summaries. The interfaces present for each item: document title; a text snippet, that is a short text extract from the source document that closely relates the query terms; the URL; and (apart from text interface) a visual summary component.

In order to control the design of the interfaces, a template was created to enable data for the five interfaces to be consistently and uniformly added. On the template, the text summaries are shown on the right-hand side at the screen, and pictures are displayed at a maximum of 200x150 pixels using original ratio on the left side. For each particular topic, the same textual surrogates (titles, snippets and URLs) are shown in exactly the same place and using the same format on all of the interfaces. The visual summaries are also displayed in exactly the same place on each interface, except for the text-only interface where the visual summaries are replaced by white space. Figure 1 shows an example of the salient image interface for one of the topics involved in this study. So, for all the interface that present the same topic, text summary region (A) was the same, and only the visual summary region (B) was changed.

Apart from the text-only interface, the remaining interfaces provide a specific type of visual summary on their search results presentation for each result page. The four visual summaries are: thumbnails, visual tags, visual snippets, and salient images. Some of these visual summaries have been evaluated by other researchers.

Thumbnail: A thumbnail is a miniature snapshot of a web page. An example is shown in Figure 2(a). A software tool called WebShot[1] was used to generate the screenshots for the test collection, to ensure the same properties for all the thumbnails.

Visual tags: The second visual summary is our approach, the visual tags summary, which is a combination of a thumbnail with a tag cloud of the retrieved web page. A tag cloud presents the most frequently used words in a document and shows the relative importance of terms using different font sizes. Our hypothesis is that this combined visual tags will provide an effective cue to the content of the retrieved web page. The construction of the visual tags includes two main stages. Firstly, a transparent image of each tag cloud was created using Wordle website[2]. The next step was to combine this with the thumbnail of the related web page. Buscher et al. [5] have found that people focus more on the top left corner of a web page, because the logo and the main navigation bar are usually located in that area, so to preserve this information region the tag cloud was located on the right of the thumbnail as shown in Figure 2(b).

Visual snippet: Visual snippets were proposed by Teevan et al. [19], and consist of a logo, a salient image and the page title. In this experiment, the visual snippet is the integration of the salient image from the retrieved web page and the website logo as shown in Figure 2(c). The page title was not included in our visual snippet because it is already presented in the related textual surrogates. Salient images were collected using Google image search over the target URL, and selecting the top-ranked image.

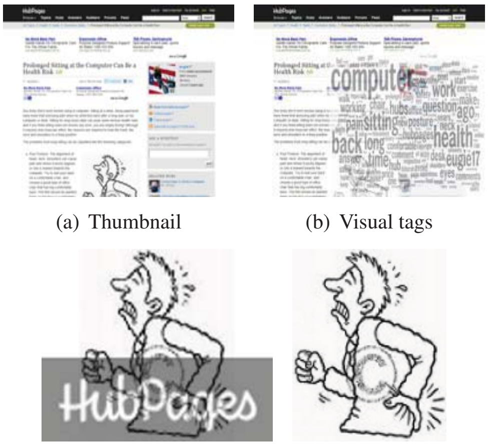
Salient image: The fourth type of visual summary is a salient image extracted from the underlying web page, see Figure 2(d). The salient image is extracted using Google image search, as explained for visual snippets.

3.3 Topic selection

Web search tasks can be classified as informational, transactional or navigational [5]. In this study, we focus on informational search tasks which aim to find specific information for a given topic. Five informational topics on general knowledge were developed :

[1] www.websitescreenshots.com

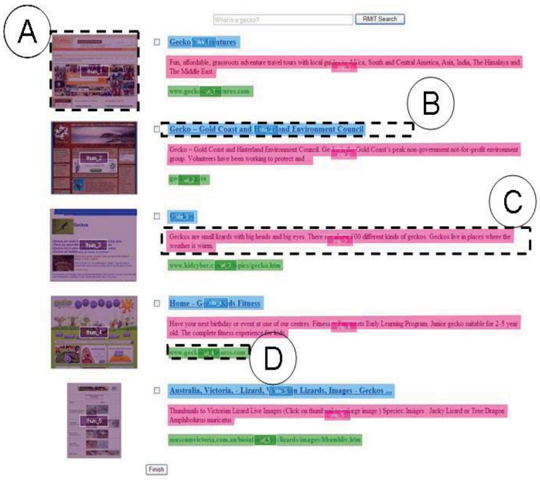
[2] www.wordle.net



(a) Thumbnail (b) Visual tags

(c) Visual snippet (d) Salient image

Figure 2: Examples of the form types of visual summaries.



□ Gecko Adventures

Fun, affordable, grassroots adventure travel tours with local guides in Africa, South and Central America, Asia, India, The Himalaya and The Middle East.

www.geckoadventures.com

□ Gecko – Gold Coast and Hinterland Environment Council

Gecko – Gold Coast and Hinterland Environment Council. Gecko is the Gold Coast's peak non-government not-for-profit environment group. Volunteers have been working to protect and ...

□ Geckos are small lizards with big heads and big eyes. There are about 700 different kinds of geckos. Geckos live in places where the weather is warm.

www.kidcyber.c... pics/gecko.htm

□ Home - Gecko Kids Fitness

Have your next birthday or event at one of our centres. Fitness Program meets Early Learning Program. Junior gecko suitable for 2-5 year old. The complete fitness experience for kids.

□ Australia, Victoria, - Lizard, Victorian Lizards, Images - Geckos ...

Thumbnails to Victorian Lizard Live Images (Click on thumbnail to enlarge image.) Species: Images . Jacky Lizard or Tree Dragon Amphibolurus muricatus

museumvictoria.com.au/bioinf... lizards/images/thumbliv.htm

Figure 3: The mask used to collect time spent on the specific informative components (A) Exact visual summary. (B) Page title. (C) Text snippet. (D) URL.

1. What are the side effects of energy drinks?
2. What is a gecko?
3. What is an appropriate sitting posture at a computer?
4. What is a solar eclipse?
5. What is a Vuvuzela?

To obtain realistic search engine results, the top ten search results from the Bing search engine were selected for each query. Wikipedia entries were excluded as they have obvious answers for the experimental tasks, then five items were randomly chosen from the remaining search results. To ensure balance in the result sets, the quality of the five selected items was restricted to include at least one relevant and one non-relevant answer, as judged by the authors.

3.4 Size of the visual summaries

Based on studies by Kaasten et al. [13] and Won et al. [20], the visual summaries in our study were set to a size of 200x150 pixels. A pilot test showed this thumbnail size to be appropriate. Search engines such as Google and Yahoo usually present ten items per page for query results, so presenting visual summaries at 200x150 pixels would not require more space than the size of a standard search result page.

3.5 Experiment design

After reading a topic from the screen, the participant clicked a "Start" button to load the search interface. Five items were displayed as search results, and participants were asked to select all items that they consider to be a relevant answer for the task. Then, the participant clicked on a "Finish" button to move to the next task.

The presentation of topics and interfaces were determined by a Latin square, giving 25 combinations of interfaces and topics, to control for presentation order effects.

3.6 Measures

User behavior is analysed by calculating the average time that users spent on specific informative components. A mask was built in the eye tracking data to collect these areas of attention, as shown in Figure 3.

The effectiveness of visual summaries was measured by click precision, click recall, and click F-measure. Click precision measures the correctly identified relevant answers as a proportion of all answers that the users selected, while click recall shows the number of relevant answer selected by users as a proportion of the total number of relevant answer available for that topic. The click F-measure gives the harmonic mean between click precision and click recall. Also, scripts were built into the HTML documents to record interaction events: the time it takes to identify a relevant item, the number of click events, and the total time taken to finish the task. In addition, t-test (p) and Chi squared test (χ^2) are applied to find the statistical difference between interfaces on each measure.

4 Results

We analyse user behavior when carrying out the five informational search topics, using a different interface for each, based on topic completion time and the relative attention paid to different summary features (page title, textual snippet, URL and visual summary).

4.1 Effectiveness of relevance prediction

In the user study, participants were asked to select all answer items that looked relevant for the given search topic. Table 2 shows the click precision, click recall and click F-measure for how effectively the users were able to identify relevant answers. We treat the text-only

Answer	Text	Thum	Tag	Image	Visual
Relevant	71	71	64	64	72
Non-relevant	18	26	34	21	27

Table 1: Distribution of the number of relevant and non-relevant answers selected by users, grouped by interface.

interface as a baseline, and compare the performance of each form of visual summary system against this using a t-test. Click precision shows no significant difference between the visual summary interfaces compared to the text interface, except for the visual tags interface which is significantly worse ($p = 0.005$). These observations indicate that the tag cloud can mislead users, causing non-relevant items to appear as potentially useful. Although the salient image interface achieves the highest average click recall (0.6), a statistical test shows no significant difference to the text-only interface ($p = 0.753$).

The number of relevant and non-relevant items that users selected are shown in Table 1, split by the interface used. The results show largely consistent rates of success with no significant differences between the interfaces ($\chi^2, p > 0.1$).

4.2 Interaction with textual summaries

User interaction with the search results was captured using eye tracking data. By using this dataset, we can investigate how users interact with textual summaries when additional visual summaries are presented. The amount of time spent looking at the overall text and visual summary regions is shown in Table 3. Users spent substantially more time looking at the text region for all interfaces. While users in general spent less time looking at the text region when using a visual summary interface, this difference was only significant for the interface that presented salient images (p = 0.032).

At the component level, user attention to specific informative components was evaluated by collecting the amount of time that the user's gaze rested on each component. The gaze regions were closely bounded on the interface component, leaving regions of white-space between them, as shown in Figure 3.

Table 3 shows the proportion of viewing time that users spent looking at the specific informative components (title, textual snippet, URL, and visual summary). Using the text interface as a baseline, there is no significant difference to the amount of time spent looking at titles or URLs when presented with a visual summary. However, on all the interfaces, users spent more time looking at page title components compared to URLs. Interestingly, the amount of time spent viewing textual snippets was significantly less when any visual summaries were displayed (for all the visual interfaces p <0.05). That is, presenting the visual summary feature decreased the attention that users gave to the text-based summary information.

Figure 4 shows the percentage of time spent on the four specific informative components. On all the interfaces, users spent more time looking at page title and textual snippets than visual summaries and URLs. Comparing the four visual summaries with each other indicates that users spent more time looking at visual tags than other visual summaries, presumably because they spent time on reading the text of the visual tags.

We also analysed how users scanned the search results with each interface by collecting the time that users spent looking at each informative component for each search result item. The results show that users were more influenced by the vertical list of the search results when they used the text interface, but this behavior was less apparent on the interfaces that present visual summaries as shown in Figure 5. This supports our observation that users spend significantly less time looking at textual snippets when using the visual interfaces. This behavior appears to be one of the main reasons that users are misled when trying to identify relevant answers for the informational tasks.

4.3 Interaction with the visual search interfaces

Next we study user attention in relation to the different informative components of the result screen. A broader comparison was conducted, by pooling the data for the four interfaces that include visual summaries, and then comparing the four attention areas (informative components) for those. The results show that users spent significantly more time looking at the textual snippets than the other informative components (p <0.001). However, there is no statistically significant difference between time spent on page titles and visual summaries, based on the aggregated attention areas.

4.4 Overall task completion time

The performance of users to complete a task was evaluated by collecting: the time users spent on each task; the time taken to first selection; and, the time taken to select first relevant item. However, no statistically significant differences were found between the five interfaces. Table 4 shows the average time spent to answer the search tasks for each interface. Although users required the least amount of time to finish their search tasks with the salient image interface, there was no statistically significant improvement compared to using the text-only interface ($p = 0.503$). Also, we calculated the average time that a user spent to answer each search task for each interface. However, no significant difference was observed between the interfaces on the measures of time completion.

5 Discussion and Conclusion

In this study, we evaluated the impact of different types of visual summaries on user behavior and performance. Fifty participants carried out five informational topics

Measures		Text	Thum	Tag	Visual	Image
Click Precision	Average	0.865	0.794	0.689	0.800	0.820
	Stdev	0.237	0.270	0.365	0.322	0.298
	t-test		0.168	**0.005**	0.253	0.412
Click Recall	Average	0.582	0.600	0.505	0.592	0.535
	Stdev	0.263	0.316	0.320	0.335	0.303
	t-test		0.753	0.193	0.868	0.413
Click F-measure	Average	0.645	0.625	0.545	0.624	0.569
	Stdev	0.190	0.232	0.290	0.271	0.250
	t-test		0.639	**0.043**	0.653	0.267

Table 2: Click precision, click recall and click F-measure for user selection of search result items.

Measures		Text	Thum	Tag	Visual	Image
Text summary region	Average	23.853	19.684	19.205	19.150	16.923
	Stdev	16.504	14.161	13.976	14.424	15.318
	t-test		0.178	0.132	0.132	**0.032**
Visual summary region	Average	0.113	3.742	5.771	4.436	3.919
	Stdev	0.244	4.038	6.068	4.650	4.294
	t-test		<0.001	<0.001	<0.001	<0.001
Titles	Average	4.263	3.514	3.846	3.791	3.615
	Stdev	2.176	2.369	3.032	3.292	2.944
	t-test		0.103	0.432	0.400	0.214
Textual snippets	Average	9.512	6.245	5.866	5.931	5.777
	Stdev	8.585	5.313	6.250	5.692	7.516
	t-test		**0.024**	**0.017**	**0.016**	**0.023**
URLs	Average	0.943	0.700	0.632	0.676	0.606
	Stdev	1.493	0.898	0.880	0.887	0.797
	t-test		0.327	0.209	0.280	0.163
Exact visual summaries	Average	-	0.659	1.136	0.744	0.645
	Stdev	-	0.754	1.186	0.801	0.728

Table 3: The Average, Standard deviation and *t*-test for the time in seconds spent viewing summary regions and informative components. The *t*-test evaluates the difference between the text interface and each of the interfaces that includes a visual component.

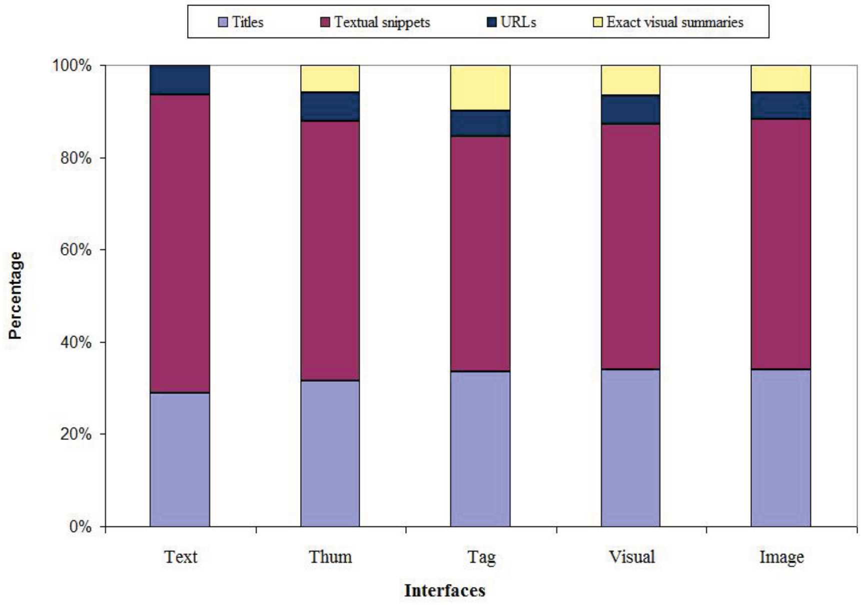


Figure 4: The percentage of time spent on the specific informative components.

Measures		Text	Thum	Tag	Visual	Image
Time to finish	Average	27.507	26.839	29.312	27.629	25.117
	Stdev	17.542	17.236	17.537	18.394	18.021
	t-test	N/A	0.848	0.608	0.973	0.503

Table 4: The Average time spent to finish search tasks for each interface. The t-test compares the text-only interface with each of the visual interfaces.

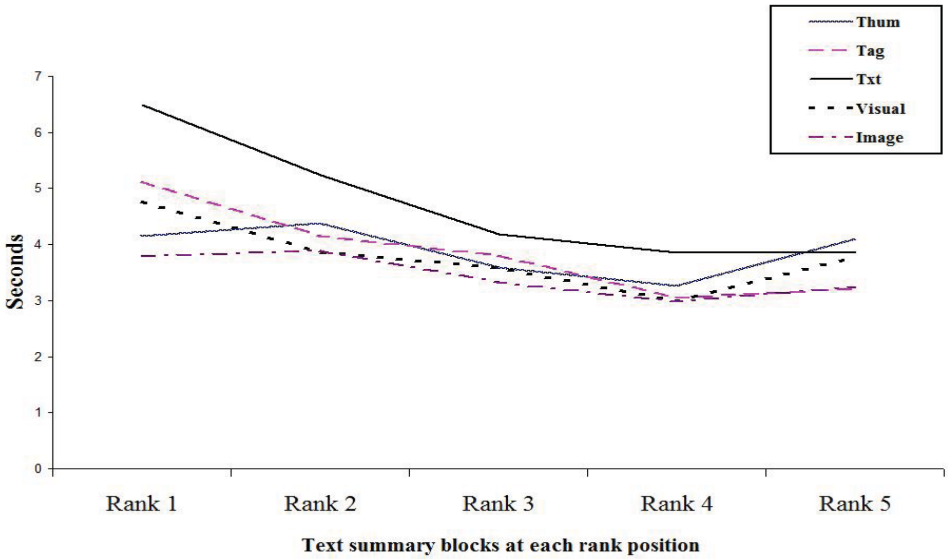


Figure 5: Average time spent on the textual surrogates for the five search result items.

using five different interfaces. Our study primarily focused on evaluating the ability of users to predict the relevance of answers when visual summaries are provided. Other studies [11, 19] focus on evaluating visual summaries in terms of finding and refinding issues, whereas in our study we considered informational search tasks.

Providing additional visual summary information with the text search results did not significantly improve the ability of users to predict the relevance of a result page for an informational search task. Although the salient image interface achieved the highest average click recall, a statistical test showed no significant difference compared to the text-only interface baseline. Also, no significant difference was found for the number of relevant result pages that users selected for each interface. Further, the results show that adding visual summaries may mislead users to select non-relevant results pages for the search topics. A possible reason for explaining this behavior is that users are not as familiar with these novel approaches as with standard text-only result lists.

We studied user behavior when presented with an additional visual summary, and the results show that visual summaries significantly affect user behavior. Although the informational search tasks seem to require reading text more than looking at pictures, users in general spent less time looking at the text region when using a visual summary interface. This may also explain the lower performance when predicting the relevance of answers items when visual information is displayed.

Furthermore, we analysed how user attention is devoted to specific informative interface components. Users spend more time looking at tag clouds, but there is no significant difference in attention between the four interfaces that include visual information. Also, the results show that users scan the search results exhaustively for the text interface, but economically for the visual interfaces. With the text interface, users spent more time looking at the top items and this amount gradually decreased as they move down the ranked list, while for the visual interfaces, the amount of time per item shows less variation.

In addition, we collected the time users spent on each task, time taken to first selection and time taken to select first relevant item. However, given our sample size no statistically significant differences were found. This suggests that visual summaries do not provide enough information for informational search tasks, since the answers for this type of search are more likely to be located in the text rather than visual summaries.

In future work, we plan to use the collected data to improve the use of an eye tracker for the evaluation of web search interfaces. Also, we plan to conduct further user studies over a wider range of tasks.

References

[1] Hilal Al Maqbali, Falk Scholer, James A. Thom and Mingfang Wu. Do users find looking at text more useful than visual representations? a comparison of three search result interfaces. In *The Fourteenth Australasian Document Computing Symposium (ADCS)*, pages 35–42, Australia, 2009.

[2] Hilal Ali[Al Maqbali], Falk Scholer, James A. Thom and Mingfang Wu. User interaction with novel web search interfaces. In *OZCHI '09: Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group*, pages 301–304. ACM, 2009.

[3] Anne Aula, Rehan M. Khan, Zhiwei Guan, Paul Fontes and Peter Hong. A comparison of visual and textual page previews in judging the helpfulness of web pages. In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 51–60. ACM, 2010.

[4] Scott Bateman, Carl Gutwin and Miguel Nacenta. Seeing things in the clouds: the effect of visual features on tag cloud selections. In *HT '08: Proceedings of the Nineteenth ACM conference on Hypertext and hypermedia*, pages 193–202. ACM, 2008.

[5] Andrei Broder. A taxonomy of web search. *SIGIR Forum*, Volume 36, Number 2, pages 3–10, 2002.

[6] Andy Cockburn, Carl Gutwin and Jason Alexander. Faster document navigation with space-filling thumbnails. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 1–10. ACM, 2006.

[7] Veronika Coltheart. *Fleeting Memories: Cognition of Brief Visual Stimuli*. MIT Press: Cambridge, 1999.

[8] Anna Divoli, Michael A. Wooldridge and Marti A. Hearst. Full text and figure display improves bioscience literature search. *PLoS ONE*, Volume 5, Number 4, pages e9619, 04 2010.

[9] Susan Dziadosz and Raman Chandrasekar. Do thumbnail previews help users make better relevance decisions about web search results? In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 365–366. ACM, 2002.

[10] May Eric, Eric Z. Ayers and John T. Stasko. Using graphic history in browsing the world wide web. In *the International World Wide Web Conference*, pages 11–14, 1995.

[11] Binxing Jiao, Linjun Yang, Jizheng Xu and Feng Wu. Visual summarization of web pages. In *SIGIR '10: Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 499–506. ACM, 2010.

[12] Hideo Joho and Joemon Jose. A comparative study of the effectiveness of search result presentation on the web. In *Advances in Information Retrieval*, Volume 3936 of *Lecture Notes in Computer Science*, pages 302–313. Springer Berlin / Heidelberg, 2006.

[13] Shaun Kaasten, Saul Greenberg and Christopher Edwards. How people recognize previously seen web pages from titles, urls and thumbnails. In *People and Computers XVI (Proceedings of Human Computer Interaction)*, pages 247–265, 2001.

[14] Zhiwei Li, Shuming Shi and Lei Zhang. Improving relevance judgment of web search results with image excerpts. In *WWW '08: Proceeding of the 17th international conference on World Wide Web*, pages 21–30. ACM, 2008.

[15] Maarten, Mary P. Czerwinski, Maarten Van Dantzich, George Robertson and Hunter Hoffman. The contribution of thumbnail image, mouse-over text and spatial location memory to web page retrieval in 3d. In *Human-Computer Interaction – INTERACT'99*, pages 163–170. IFIP, 1999.

[16] William Ogden, Mark Davis and Sean Rice. Document thumbnail visualizations for rapid relevance judgments: When do they pay off? In *The Seventh Text REtrieval Conference (TREC7). NIST*, pages 528–534, 1998.

[17] Johann Schrammel, Michael Leitner and Manfred Tscheligi. Semantically structured tag clouds: an empirical evaluation of clustered presentation approaches. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, pages 2037–2040. ACM, 2009.

[18] James Sinclair and Michael Cardew-Hall. The folksonomy tag cloud: when is it useful? *J. Inf. Sci.*, Volume 34, Number 1, pages 15–29, 2008.

[19] Jaime Teevan, Edward Cutrell, Danyel Fisher, Steven M. Drucker, Gonzalo Ramos, Paul André and Chang Hu. Visual snippets: summarizing web pages for search and revisitation. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, pages 2023–2032. ACM, 2009.

[20] Sungjoon Steve Won, Jing Jin and Jason I. Hong. Contextual web history: using visual and contextual cues to improve web browser history. In *CHI '09: Proceedings of the 27th international conference on Human factors in computing systems*, pages 1457–1466. ACM, 2009.

[21] Allison Woodruff, Andrew Faulring, Ruth Rosenholtz, Julie Morrsion and Peter Pirolli. Using thumbnails to search the web. In *CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 198–205. ACM, 2001.

[22] Songhua Xu, Tao Jin and Francis C. M. Lau. A new visual search interface for web browsing. In *WSDM '09: Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 152–161. ACM, 2009.

Efficient Accumulator Initialisation

Xiang-Fei Jia
Department of
Computer Science
University of Otago
Dunedin, New Zealand
fei@cs.otago.ac.nz

Andrew Trotman
Department of
Computer Science
University of Otago
Dunedin, New Zealand
andrew@cs.otago.ac.nz

Richard O'Keefe
Department of
Computer Science
University of Otago
Dunedin, New Zealand
ok@cs.otago.ac.nz

Abstract *IR efficiency is normally addressed in terms of accumulator initialisation, disk I/O, decompression, ranking and sorting. Traditionally, the performance of search engines is dominated by slow disk I/O, CPU-intensive decompression, complex similarity ranking functions and sorting a large number of candidate documents. However, after we have applied a number of optimisation techniques, our search engine is bottlenecked by accumulator initialisation. In this paper, we propose an efficient accumulator initialisation algorithm, which represents the traditional static accumulator array as a logical two dimensional table and uses a number of flags to track the initialisation status of the accumulators. The efficiency of the algorithm is verified by a simulation program and a search engine. The overall performance can be as good as a 93% increase in throughput.*

Keywords Accumulator Initialisation, Efficiency, Postings Pruning.

1 Introduction

Effectiveness and efficiency are two of the main issues in Information Retrieval (IR). Effectiveness has been the main focus of research. In recent years, efficiency has started to draw more attention under the trend of larger document collection sizes.

IR efficiency is normally addressed in terms of accumulator initialisation, disk I/O, decompression, ranking and sorting. A large portion of the performance of search engines is dominated by (1) slow disk read of dictionary terms and the corresponding postings lists, (2) CPU-intensive decompression of postings lists, (3) complex similarity ranking functions and (4) sorting a large number of possible candidate documents. The effect of accumulator initialisation on the performance has almost been ignored.

However, after we applied a number of optimisation techniques, our search engine was bottlenecked by accumulator initialisation. We have deployed space efficient compression algorithms for storing the dictionary and inverted files, disk I/O is completely eliminated by simply storing the index in memory. Instead of storing the traditional ⟨document number, term frequency⟩ pair in postings, we pre-compute and store impact values instead [15, 1]. The search engine simply adds the impact values when ranking.

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010. Copyright for this article remains with the authors.

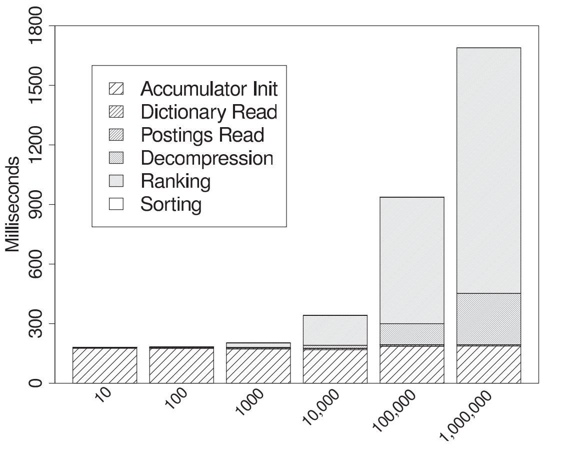


Figure 1: The performance of our optimised search engine using traditional static array for accumulator initialisation.

Postings pruning at query time is a very effective method for reducing the number of postings to be processed and the number of accumulators to be sorted, while still maintaining high precision [8, 14, 23, 17, 1, 22]. Since only part of the postings lists is processed in pruning, only partial decompression of postings lists is required [14, 13, 3, 2]. Our search engine has adopted a heap data structure to keep track of the top k documents and static pruning of postings lists with partial decompression.

Figure 1 shows the performance of our optimised search engine with these optimisations enabled. The document collection and queries are the INEX 2009 Wikipedia collection [18] and the 115 Type-A (short) queries from the INEX 2009 Efficiency Track [19]. Only the top k=15 results are returned and each column in the figure corresponds to a static pruning of 10, 100, 1000, 10 000, 100 000, 1 000 000 postings. When no more than 10 000 postings are processed, the accumulator initialisation takes most of the time, between 50% and 96% of the total time. When there are 100 000 and 1 000 000 postings (which is one third of the collection size) being processed, the accumulator initialisation still takes 20% and 11% of the total evaluation time respectively.

In this paper, we propose an efficient accumulator initialisation algorithm, using static data structures, for the *term-at-a-time* approach. The algorithm logically partitions the static array of accumulators into a two dimensional table. A flag is created for each logical row to indicate if that row has been initialised. Before processing a new query, only the flags are re-initialised instead of re-initialising all accumulators. Because the algorithm keeps track of all accumulators, there is no loss of precision.

The remainder of the paper is organised as follows. In Section 2, we discuss the related work. Second 3 describes in detail how the algorithm works and presents a mathematical model. In Section 4, the performance of the algorithm is conducted on a simulation and our search engine. Section 5 concludes.

2 Related Work

Disk I/O involves reading query terms from a dictionary (a vocabulary of all terms in the collection) and the corresponding postings lists for the terms. The dictionary has a small size and can be loaded into memory at start-up. However, due to their large size, postings are usually compressed and stored on disk. A number of compression algorithms have been developed and compared [21, 4]. Another way of reducing disk I/O is caching, either at application level or system level [5, 11]. Since the advent of 64-bit machines with vast amounts of memory, it has become feasible to load both the dictionary and the compressed postings into main memory, thus eliminating all disk I/O. Reading both dictionary and postings lists into memory is the approach taken in our search engine.

The processing (decompression and similarity ranking) of postings and subsequent sorting of accumulators can be computationally expensive, especially when queries contain frequent terms. Processing of these frequent terms not only takes time, but also has little impact on the final ranking results. Postings pruning at query time is a method to eliminate unnecessary processing of postings and thus reduce the number of nonzero accumulators to be sorted. A number of pruning methods have been developed and proved to be efficient and effective [8, 14, 23, 17, 1, 22]. In previous work [22], the *topk* pruning algorithm partially sorts the static array of accumulators using a special version of quick sort [6] and statically prunes postings. Based on this work, we have developed the *heapk* pruning algorithm. Instead of explicitly sorting the accumulators, we uses a heap data structure to keep track of the top documents.

Traditionally, postings are stored in pairs of ⟨document number, term frequency⟩ pairs. However, postings should be impact ordered so that most important postings can be processed first and the less important ones can be pruned using pruning methods [16, 17, 1]. One approach is to store postings in order of term frequency and documents with the same term frequency are grouped together [16, 17]. Each group stores the term frequency at the beginning of the group followed by the compressed differences of the document numbers. The format of a postings list for a term is a list of the groups in descending order of term frequencies. Another approach is to pre-compute similarity values and use these pre-computed impact values to group documents instead of term frequencies [1]. Pre-computed impact values are positive real numbers. In order to better compress these numbers, they are quantised into whole numbers [15, 1]. Three forms of quantisation method have been proposed (*Left.Geom*, *Uniform.Geom*, *Right.Geom*) and each of the methods can better preserve certain range of the original numbers [1]. In our search engine, we use pre-computed BM25 impact values to group documents and the differences of document numbers in each group are compressed using Variable Byte Coding by default. We choose to use the *Uniform.Geom* quantisation method for transformation of the impact values, because the *Uniform.Geom* quantisation method preserves the original distribution of the numbers, thus no decoding is required at query time. Each impact value is quantised into an 8-bit whole number.

Since only partial postings are processed in query pruning, there is no need to decompress the whole postings lists. Skipping [14] and blocking [13] allow pseudo-random access into encoded postings lists and only decompress the needed parts. Further research work [3, 2] represent postings in fixed number of bits, thus allowing full random access. Our search engine partially decompress postings list based on the worst case of the static pruning. Since we know the parameter value of the static pruning and the biggest size of a uncompress impact value (4 bytes), we can multiply these number together to find the cut point for decompression. We can simply hold decompression after that number of postings have been decompressed.

A number of accumulators, usually as a static array, need to be created and initialised for *term-at-a-time* processing [8, 10]. The accumulators hold the intermediate accumulated results for each document. For large collections, a large number of accumulators has to be used. Initialisation of large number of accumulators can take time. One solution to cut down the initialisation time is to use few accumulators, which are allocated using dynamic search structures [15, 14]. Depending on which dynamic structure is used, the memory space required for each accumulator can be several times that of the static array structure. For example, a balanced Red-Black tree structure [9] requires about 20 bytes for each accumulator on 32-bit architectures, and about

32 bytes on 64-bit architectures, compared with only 4 bytes needed in a static array. Only 20% (12.5% for 64-bit) or less of the total number of accumulators should be allocated, otherwise the dynamic structure uses more memory. The more memory is used, the longer it takes to allocate.

Since only a portion of accumulators can be allocated using dynamic search structures, a pruning algorithm has to be used to keep only the top candidates and to prune other less important ones [15, 14]. On the other hand, our search engine allocates all accumulators, and not only keeps track of the top candidates but also updates the less important accumulators. This leaves the possibility for the less important candidates to be among the top ones at the final stage.

The criticism of the *term-at-a-time* approach is the requirement of accumulators in order to hold intermediate results. Alternatively, the *document-at-a-time* approach ranks one document at a time, thus does not need to hold intermediate results [24, 20]. However, the *document-at-a-time* approach requires random scan of postings lists, which takes time [20].

3 The Algorithm

Instead of using dynamic accumulator structures, we use two static arrays. One array is used to hold all the accumulators (one for each document), the other to hold a number of flags. Every flag is associated with a particular subset of the accumulators, indicating the initialisation status for that set of accumulators. Essentially, we turn the one dimensional array of accumulators into a logical two dimensional table as shown in Figure 2. The dimension of the table is represented by $height$ and $width$. The number of flags is the same as the height of the table.

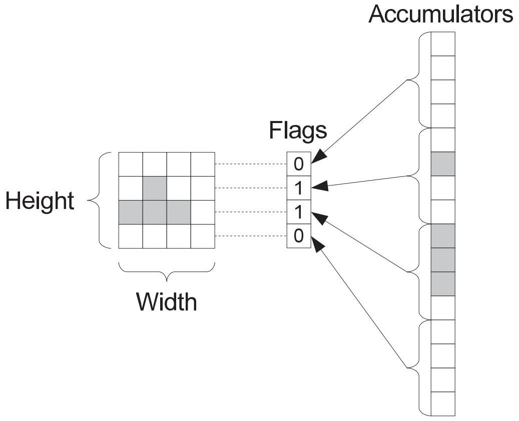


Figure 2: The representation of the accumulators in a logical two dimensional table.

One obvious question is why not just dynamically allocate each row only when needed and replace the flags with pointers pointing to the dynamically allocated rows. There are two efficiency issues when rows are dynamically allocated. First, it is faster to allocate a large chuck of memory in one go, rather than splitting the same of amount of memory into many smaller pieces and allocating one piece at at time. Modern computers come with very large memories, so it is worth to sacrifice small amount of memory for speed. Second, two steps are required to locate each accumulator because rows are not guaranteed to be allocated consecutively in memory (the first step is to locate the row and the second step to find the offset in the row).

Another question is how to determine the dimension of the logical table. In the following sections, a mathematical model is provided to answer this question formally and a simulation program is tested with various sizes of width. For now, let us concentrate on how the algorithm works.

Initially, the flags are initialised to zero, indicating all the accumulators having zero values. When updating an accumulator with a new value, the flag associated with that row of the accumulator is set to 1. For the example shown in Figure 2, The second accumulator in the second logical row has a non-zero value and the associated flag for the row has a value of 1.

The width has to be a whole number at least 2. The height can be calculated according to the width and the size of the document collection, as shown in Algorithm 1. Because the total number of accumulators represented by the logical table can be more than the collection size, extra accumulators (shown as $padding$ in Algorithm 1) are allocated in the accumulator array (this is for efficiency). The number of extra accumulators required is usually small and the worst case is $width - 1$.

Algorithm 1 Accumulator Initialisation

```
Require: $width \geq 2$
$N \leftarrow total\_documents\_in\_collection$
$height \leftarrow (N/width) + 1$
$init\_flags \leftarrow$ new $array[height]$
initialise $init\_flags$
$padding \leftarrow (width * height) - N$
$acc \leftarrow$ new $array[N + padding]$
```

Algorithm 2 Accumulator Update

```
Require: $doc\_id \geq 0$ and $doc\_id < N$
$row \leftarrow doc\_id/width$
if $init\_flags[row] == 0$ then
    $init\_flags[row] \leftarrow 1$
    initialise the row of the accumulators in $acc$
end if
$acc[doc\_id] \leftarrow acc[doc\_id] + new\_rsv$
```

Algorithm 2 shows how to update the accumulators. First, the logical row of an accumulator can be obtained by a division operation of the index of the accumulator. Second, the row flag is checked. Two possible cases can happen. If the flag is 0, it set to 1, the associated accumulators are initialised, and the new value is added to the accumulator. If the flag is 1, the new value can be simply aggregated to the accumulator.

3.1 The mathematical model

Let D be the number of documents, Q the number of terms in the query, L the size of the postings list for each term, B the width of each row, $K = D/B$ the number of rows, and U the number of rows that are not used. Zipf's law tells us that postings lists vary enormously in size, but we are considering a system with pruned impact-ordered lists and pessimistically assume all postings lists are of the pruned length L.

Processing the query is a many step process that starts with clearing the row flags. Then for each term, the postings are loaded, decompressed, and processed. That processing in turn involves computing the row number and checking the row flag, if the flag is zero the row is cleared and the flag is set. The accumulator is always increased (and top-k processing is done). At the end of the query the top-k accumulator pointers are sorted.

For only two of these steps does the cost depend on the width and height of the two-dimensional accumulator table:

- c_1K, zeroing the flags for each row;
- $c_2(K - U)B$, zeroing all accumulators in all flagged row (and setting the bits);

where c_1 and c_2 are unknown constant factors. We wish to minimise the cost of initialising the flags, c_1K, plus the cost of initialising all the used rows, $c_2(K - U)B$.

In practice, some documents are more likely to be selected than others (due to the clustering hypothesis), and the user does not expect the terms to be independent. That is, they expect fewer than QL distinct documents to be found. We pessimistically assume each term is independent and identically (randomly) distributed for this analysis.

The postings for a term would normally be sampled without replacement, but if L is big enough and L/D is small enough, that can be approximated by sampling with replacement.

P(row k is unused)
= P(document $d \subseteq kB \ldots kB + B - 1$ not chosen)
= P(document d is not chosen)^B
= P(document d is not in postings for term $t \subseteq t_1..t_Q$)^B
= P(document d is not in postings for term t)^{QB}
= $(1 - L/D)^{QB}$

The postings are randomly distributed in the postings list and so represent independent trials, there are K rows and the probability of a single row being unused is $(1 - L/D)^{QB}$. So the distribution of U is Binomial$(K, (1 - L/D)^{QB}$. As mean(Binomial(n,p)) $= np$, the mean of the number of unused rows is $K(1 - L/D)^{QB}$. It depends on the number of terms in the query and the number of postings for each term.

We wish to minimise $c_1K + c_2(K - U)B$. Since $B = D/K$ we get

$$\begin{aligned} & c_1K + c_2(K-U)B \\ = \; & c_1K + c_2\tfrac{(K-U)D}{K} \\ = \; & c_1K + c_2\tfrac{KD-UD}{K} \\ = \; & c_1K + c_2\tfrac{KD}{K} - c_2\tfrac{UD}{K} \\ = \; & c_1K + c_2D - c_2\tfrac{UD}{K} \end{aligned}$$

Recall that U is Binomial$(K, (1 - L/D)^{QB}$ and the mean value is $K(1 - L/D)^{QB}$. We get

$$\begin{aligned} & c_1K + c_2D - c_2\tfrac{UD}{K} \\ = \; & c_1K + c_2D - c_2\tfrac{K(1-\frac{L}{D})^{BQ}D}{K} \\ = \; & c_1K + c_2D - c_2(1-\tfrac{L}{D})^{BQ}D \end{aligned}$$

Applying the binomial theorem to $(1 - \frac{L}{D})^{QB}$ and truncatingat the first two terms, we get

$$\begin{aligned} & c_1K + c_2D - c_2(1-\tfrac{L}{D})^{BQ}D \\ \approx \; & c_1K + c_2D - c_2(1-\tfrac{L}{D}BQ)D \\ \approx \; & c_1K + c_2D - c_2(D-\tfrac{L}{D}BQD) \\ \approx \; & c_1K + c_2D - c_2D + c_2LBQ \\ \approx \; & c_1K + c_2LBQ \\ \approx \; & c_1\tfrac{D}{B} + c_2LBQ \end{aligned}$$

Fermat's stationary point theorem tells us that a differentiable function has its maxima and minima only on the boundaries or where the first derivative is zero. The second derivative tells whether an extreme point is a maximum or a minimum.

$$\begin{aligned} & \tfrac{d}{dB}(c_1\tfrac{D}{B} + c_2LBQ) \\ = \; & \tfrac{d}{dB}(c_1\tfrac{D}{B}) + \tfrac{d}{dB}(c_2LBQ) \\ = \; & -c_1\tfrac{D}{B^2} + c_2LQ \end{aligned}$$

$$\begin{aligned} & \tfrac{d^2}{dB^2}(c_1\tfrac{D}{B} + c_2LBQ) \\ = \; & \tfrac{d}{dB}(-c_1\tfrac{D}{B^2} + c_2LQ) \\ = \; & c_1\tfrac{D}{B^3} \end{aligned}$$

By setting the first derivative to zero, we get

$$\begin{aligned} c_1\tfrac{D}{B^2} &= c_2LQ \\ B^2 &= \tfrac{c_1D}{c_2LQ} \\ B &= \sqrt{\tfrac{c_1D}{c_2LQ}} \end{aligned}$$

Since the second derivative is greater than zero, this must be a local minimum.

4 Experiments

We conducted all our experiments on a system with dual quad-core Intel Xeon E5410 2.3 GHz, DDR2 PC5300 8 GB main memory, Seagate 7200 RPM 500 GB hard drive, and running Linux with kernel version 2.6.30.

For both the simulation and our search engine, we set the width of the table to a power of 2. This allows us to look up the row efficiently using a hashing function that is just a bit shift. A shift operation is considerably faster than a division.

First, we tried a number of simulation tests to inspect the behaviour of the algorithm. Then we integrated the algorithm into our search engine and investigated its performance using the INEX 2009 Wikipedia collection [18] and queries from the INEX 2009 Efficiency Track [19].

4.1 Simulation

We wrote a simulation program to test both the traditional static array allocation and our proposed logical two dimensional table algorithm.

The program takes five parameters; (1) *collection_size*, (2) *postings_list_length*, (3) *num_of_terms*, (4) *width_in_bits* and (5) *num_of_repeats*. The first two parameters allow the program to simulate different collection sizes with various lengths of postings lists. The third parameter tries to simulate real world queries with different number of terms. The *width_in_bits* parameter sets the size of the width for the logical two dimensional table. The last parameter simply tells the program to repeat a number of times with the same settings.

Each posting only contains an index number (the document number) for indexing into the accumulators. All the index numbers are randomly generated using the Mersenne Twister 64-bit random generator [12]. Every run of the simulation is guaranteed to execute with a different seed for the random number generator and each run was repeated 20 times. For each run, the same lists of postings are used by both the accumulator initialisation techniques.

Figure 3 shows the results of simulating a collection of three million documents. The horizontal axes represents the width in bits and the vertical axes are the performance ratio of the two accumulator initialisation techniques (the total time taken by the logical two dimensional table over the total time taken by the static array structure). A ratio below 100% means the performance increase in our proposed algorithm again the static array approach and above 100% means performance decrease. Four sets of simulation were conducted, each with 1, 2, 3 and 10 terms. For each set, various lengths of postings lists were used, ranged from 10 to 3 million.

As shown in Figure 3, the four sets of simulation showed the same pattern. When the length of the postings lists was short, smaller width for the logical table showed better performance, while larger width shower better performance for long postings lists. On average, bits between 8 and 12 showed good performance for both short and long postings lists.

One thing to note is the caching effects for simulations with a small number of postings. An example of the results being affected by caching is when there are two logical rows ($width\_in\_bit$ is 1) and only 10 postings processed. The size of the static accumulator array is 6 million bytes (the collection size times the byte size of a single accumulator) and the size of the flag array is 1.5 million bytes (we used one byte for each flag). By hand calculation, the performance ratio should be more than 25% because the size of the flag array is 25% of the static accumulator array plus the initialisation of 10 rows for the 10 postings (the worst case). However, the simulation showed that the performance ratio is only about 8.5%. This further suggests that the logical two dimensional table is more efficient because the structure has a smaller memory footprint and thus can be better cached by the CPU (the CPU has 6 MB of L2 cache).

Overall, our proposed logical two dimensional table for accumulator initialisation shows good performance, especially when postings lists are short. This suggests that the algorithm can be better used together with postings pruning techniques, which only processes partial postings.

4.2 Search Engine

As discussed in Section 2, our search engine supports reading of the dictionary and postings directly in memory, impact-ordered postings, partial decompression and postings pruning at query time. In this section we discuss how the proposed accumulator initialisation algorithm performs in our search engine.

In the previous *topk* pruning implementation [22], an extra array of pointers was used and the size of the array was k. The pointers were used to keep track of the top documents in the static array of accumulators. At the final stage, the top k pointers are sorted instead of sorting the static accumulator array directly.

The implementation of our *heapk* pruning algorithm is also based on the pointer array. The *heapk* algorithm simply uses the same pointer array to keep track of top documents. The minimum accumulator among the top k is always pointed to by the first pointer in the heap. During the update of an accumulator, the new value of the accumulator is checked against the minimum. If the new value is greater than the minimum, the minimum pointer simply points to the new value and the heap structure is re-built. If the new value is less than the minimum, then there is nothing to be done. At the final stage, the documents which are tracked by the heap data structure are sorted and returned.

A modified BM25 is used for ranking. This variant does not result in negative IDF values and is defined as:

$$RSV_d = \sum_{t \in q} \log\left(\frac{N}{df_t}\right) \cdot \frac{(k_1 + 1)\, tf_{td}}{k_1 \left((1 - b) + b \times \left(\frac{L_d}{L_{avg}}\right)\right) + tf_{td}}$$

Here, N is the total number of documents, and df_t and tf_{td} are the number of documents containing the term t and the frequency of the term in document d, and L_d and L_{avg} are the length of document d and the average length of all documents. The empirical parameters k_1 and b have been set to 0.9 and 0.4 respectively by training on the INEX 2008 Wikipedia collection.

We used the INEX 2009 Wikipedia collection [18] and the 115 Type-A (short) queries for the INEX 2009 Efficiency Track [19]. The collection was indexed with

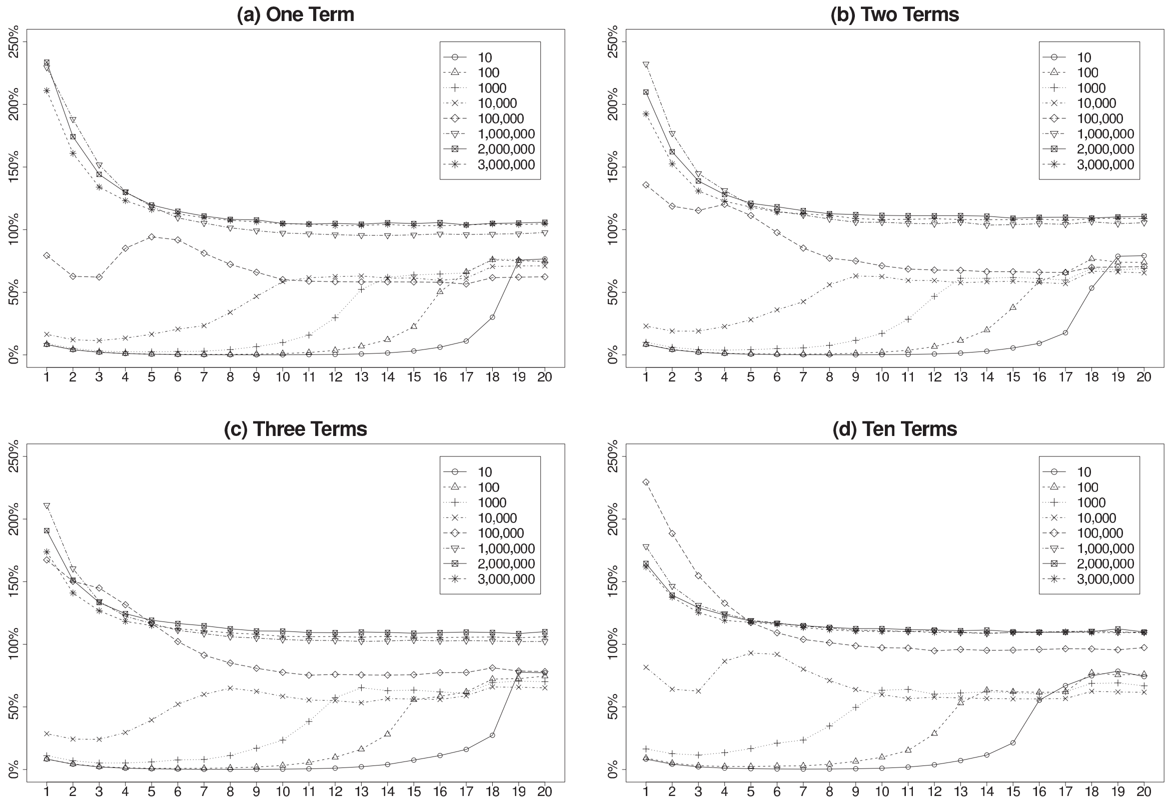


Figure 3: Simulation results on a collection of three million documents.

Collection Size	50.7 GB
Documents	2666190
Average Document Length	881 words
Unique Words	11393924
Total Words	2348343176
Postings Size	1.68 GB
Dictionary Size	269 MB

Table 1: Summary of INEX 2009 Wikipedia Collection.

no words stopped and stemming was not used. Table 1 shows a summary of the document collection.

Initially our search engine used the traditional static array approach for accumulators. To see the performance of the accumulator initialisation with regards to the overall runtime, the 115 queries were evaluated with all the optimisation options enabled (in-memory dictionary and postings, impact-ordered postings, partial decompression and the *heapk* pruning). The *heapk* pruning method was used with a value of 15 for the number of top k documents and various values of 10, 100, 1000, 10 000, 100 000, 1 000 000 for static pruning. The total run-times are shown in Figure 1. When no more than 10 000 postings were processed, the accumulator initialisation took between 50% and 96% of the total evaluation time. When 100 000 and 1 000 000 postings were processed, it took 20% and 11% respectively. As shown in previous work [22], query pruning is not only efficient, but also very effective, even when as few as 1000 postings are processed.

Now our search engine is bottlenecked by the accumulator initialisation. We need a solution for efficient accumulator initialisation. The logical two dimensional table has been integrated into the *heapk* pruning algorithm. The integration requires only changes to a few lines of code. Before processing each query, the flags in the logical two dimensional table are initialised. During the update of an accumulator, the flag is checked to see if it is necessary to initialise the logical row first.

In order to compare the performance again the static array approach, we performed the same set of tests on the logical two dimensional table. The width for the logical two dimensional table was chosen to be 8 bits. The results are shown in Figure 4(a). The only performance differences between these two structures are the accumulator initialisation, the ranking and the total times. The logical two dimensional table took about zero time for the accumulator initialisation since it was very fast to initialise 11719 flags (the height of the table calculated as shown in Algorithm 1). However, the logical two dimensional table added small amount of overhead for ranking due to the extra operations required to keep track of the flag status (as shown in Algorithm 2). When comparing the total evaluation time, the logical two dimensional table outperformed the static array structure in all runs. The best performance increase is 93% when 10 postings

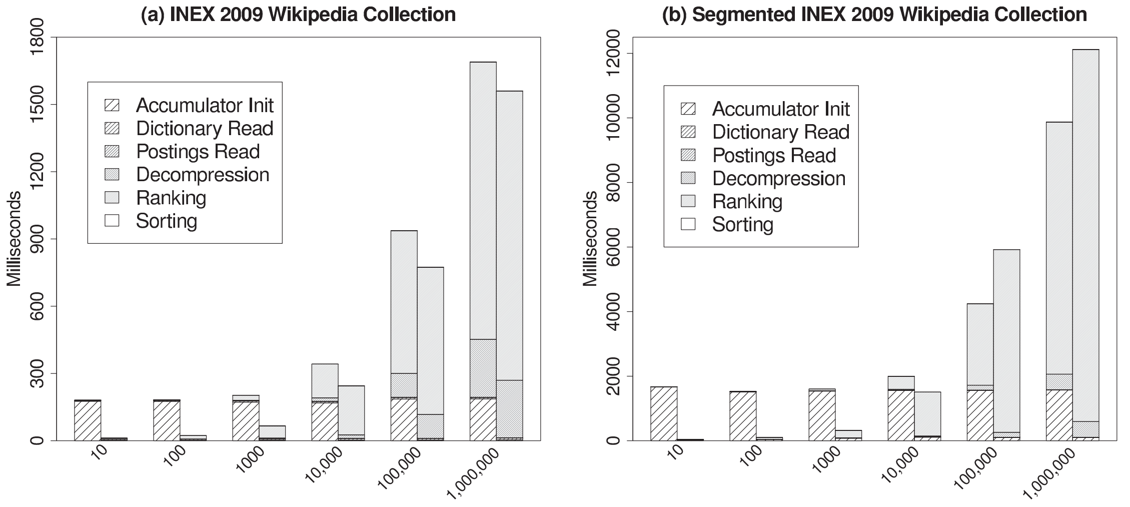


Figure 4: Comparison between the static array and logical two dimensional table structures for accumulator initialisation. The first bar in each group shows the results for the static array structure while the second shows the results for the logical two dimensional table.

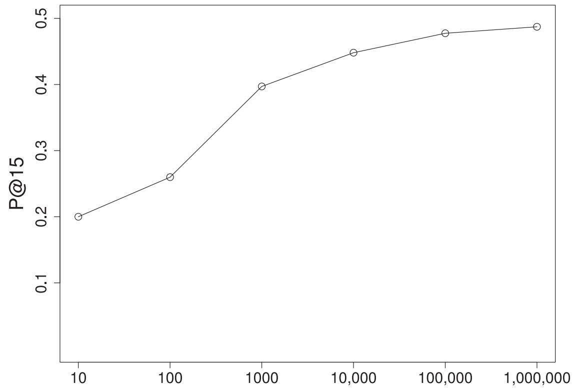


Figure 5: Precision for P@15 using assessments from the INEX 2009 Efficiency Track [19].

are processed. The performance increases for 100, 1000, 10 000, 100 000 are 86%, 67%, 28% and 17% respectively. However, there is only a 2% performance increase when 1 000 000 postings are processed.

The effectiveness of the *heapk* pruning algorithm is shown in Figure 5 using precision at 15 for various values of postings. The highest precision is 0.487 when 1 000 000 were processed. There were small precison drops 2% and 8% when 100 000 and 10 000 postings were processed respectively. The precisions were dramaticly dropped 18%, 47% and 59% when 1000, 100, 10 postings were processed. Overall, it shows that postings pruning is very effective.

We also tested the same set of experiments on a larger document collection provided by the University of Queensland. The collection has 23 million documents and is created by splitting each section of the documents as a single document in the INEX 2009 Wikipedia collection [18]. Since there is no evaluation for this collection, we cannot show precision.

Figure 4(b) shows the results. It took about 1700 milliseconds for initialising the static accumulator array, compared with zero time for the logical two dimensional table. There was a overall performance increase of 97%, 93%, 80% and 25% when 10, 100, 1000, 10 000 postings were processed respectively. However, there was a overhead of 40%, 23% for processing 100 000, 1 000 000 postings respectively.

Compared with the original Wikipedia collection, the processing of 100 000 and 1 000 000 postings caused more overhead for the logical two dimension table in the segmented collection. In the original collection, few terms have more than 1 000 000 postings. However, when documents are segmented by sections, more lists are closer to or longer than 100 000 and 1 000 000.

5 Conclusion and Future Work

In this paper we have proposed an efficient accumulator initialisation algorithm, especially when used together with postings pruning. The algorithm represents the traditional static accumulator array as a logical two dimensional table and uses an array of flags to keep track of the initialisation status of the accumulators. Before processing queries, the array of flags is initialised instead of initialising the accumulator array. During the update of accumulators, a hashing function (shift) is used to locate the accumulator.

Using two dimensional structure is not new. It has been used in other areas of Computer Science, including paging and file systems [7]. In paging and file systems, multiple dimensions (multiple level of indexing) are required in order to address large amount of memory or very big files. We will explore the use of multiple dimension structures for efficient addressing large document collections, like the ClueWeb 2009 collection (1 billion documents) in future work.

We have explained how we have integrated the logical two dimensional table into our *heapk* pruning algorithm. In future work, we will examine the efficiency of the structure in other pruning algorithms. One example is the any-time stopping pruning algorithm [1]. It uses an array of lists (indexed by quantised impact values) to keep track of the current top candidates stored in the static accumulator array. The logical two dimensional table can be integrated into this pruning algorithm without affecting the original algorithm.

For both of our simulation and search engine, we defined the width to be a power of 2 and used a shift hashing function for the logical two dimensional table. A shift hashing function is considerably faster a division. However, we have not explored how far away from the optimum ($B = \sqrt{\frac{c_1 D}{c_2 LQ}}$) this is. If the gap between the optimum and a power of 2 is very large, which means a lot un-necessary accumulators are initialised, it might be more efficient to define the width as the optimum and use division for hashing.

References

[1] Vo Ngoc Anh, Owen de Kretser and Alistair Moffat. Vector-space ranking with effective early termination. pages 35–42, 2001.

[2] Vo Ngoc Anh and Alistair Moffat. Compressed inverted files with reduced decoding overheads. pages 290–297, 1998.

[3] Vo Ngoc Anh and Alistair Moffat. Random access compressed inverted files. *Australian Computer Science Comm.: Proc. 9th Australasian Database Conf., ADC*, Volume 20, Number 2, pages 1–12, February 1998.

[4] Vo Ngoc Anh and Alistair Moffat. Inverted index compression using word-aligned binary codes. *Inf. Retr.*, Volume 8, Number 1, pages 151–166, 2005.

[5] Ricardo Baeza-Yates, Aristides Gionis, Flavio Junqueira, Vanessa Murdock, Vassilis Plachouras and Fabrizio Silvestri. The impact of caching on search engines. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 183–190, New York, NY, USA, 2007. ACM.

[6] Jon L. Bentley and M. Douglas Mcilroy. Engineering a sort function, 1993.

[7] Daniel P. Bovet and Marco Cesati. *Understanding the Linux Kernel, 3rd Edition*. O'Reilly, November 2005.

[8] Chris Buckley and Alan F. Lewit. Optimization of inverted vector searches. pages 97–110, 1985.

[9] Thomas H. Cormen, Charles E. Leiserson and Ronald L. Rivest. *Introduction to Algorithm*. The MIT Press, 1990.

[10] Donna Harman and Gerald Candela. Retrieving records from a gigabyte of text on a minicomputer using statistical ranking. *Journal of the American Society for Information Science*, Volume 41, pages 581–589, 1990.

[11] Xiang-fei Jia, Andrew Trotman, Richard O'Keefe and Zhiyi Huang. Application-specific disk I/O optimisation for a search engine. In *PDCAT '08: Proceedings of the 2008 Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies*, pages 399–404, Washington, DC, USA, 2008. IEEE Computer Society.

[12] Makoto Matsumoto and Takuji Nishimura. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Trans. Model. Comput. Simul.*, Volume 8, Number 1, pages 3–30, 1998.

[13] A. Moffat, J. Zobel and S. T. Klein. Improved inverted file processing for large text databases. pages 162–171, 1995.

[14] Alistair Moffat and Justin Zobel. Self-indexing inverted files for fast text retrieval. *ACM Trans. Inf. Syst.*, Volume 14, Number 4, pages 349–379, 1996.

[15] Alistair Moffat, Justin Zobel and Ron Sacks-Davis. Memory efficient ranking. *Inf. Process. Manage.*, Volume 30, Number 6, pages 733–744, 1994.

[16] Michael Persin. Document filtering for fast ranking. pages 339–348, 1994.

[17] Michael Persin, Justin Zobel and Ron Sacks-Davis. Filtered document retrieval with frequency-sorted indexes. *J. Am. Soc. Inf. Sci.*, Volume 47, Number 10, pages 749–764, 1996.

[18] Ralf Schenkel, Fabian Suchanek and Gjergji Kasneci. YAWN: A semantically annotated wikipedia xml corpus. March 2007.

[19] Ralf Schenkel and Martin Theobald. Overview of the inex 2009 efficiency track. In Shlomo Geva, Jaap Kamps and Andrew Trotman (editors), *Focused Retrieval and Evaluation*, Volume 6203 of *Lecture Notes in Computer Science*, pages 200–212. Springer Berlin / Heidelberg, 2010.

[20] Martin Theobald, Ralf Schenkel and Gerhard Weikum. Efficient and self-tuning incremental query expansion for top-k query processing. pages 242–249, 2005.

[21] Andrew Trotman. Compressing inverted files. *Inf. Retr.*, Volume 6, Number 1, pages 5–19, 2003.

[22] Andrew Trotman, Xiang-Fei Jia and Shlomo Geva. Fast and effective focused retrieval. In Shlomo Geva, Jaap Kamps and Andrew Trotman (editors), *Focused Retrieval and Evaluation*, Volume 6203 of *Lecture Notes in Computer Science*, pages 229–241. Springer Berlin / Heidelberg, 2010.

[23] Yohannes Tsegay, Andrew Turpin and Justin Zobel. Dynamic index pruning for effective caching. pages 987–990, 2007.

[24] Howard Turtle and James Flood. Query evaluation: Strategies and optimizations. *Information Processing & Management*, Volume 31, Number 6, pages 831 – 850, 1995.

Interaction differences in web search and browse logs

Paul Thomas
CSIRO
Canberra, Australia
paul.thomas@csiro.au

Alex O'Neill
Australian National University
Canberra, Australia
alexoneill89@gmail.com

Cecile Paris
CSIRO
Sydney, Australia
cecile.paris@csiro.au

Abstract *We use logfiles from two web servers (public and internal), two corresponding search engines, and two user populations (public and staff) to examine differences in behaviour across users and sites.*

We observe similar overall characteristics to other browsing and searching logs, but differences in behaviour between staff and the public and between external and internal sites. Staff familiarity with organisational language and structure does not translate to more effective search or navigation, although staff do expend considerable effort looking for information and often look in the wrong place. This would not be apparent from logs covering only search or only browsing behaviour.

Keywords Log analysis; user behaviour; information retrieval

1 Introduction

Transaction logs, recording users' interactions with web servers or web search engines, provide a cheap and unobtrusive source of data on how people use the web. A good deal of analysis has been carried out on log data (see e.g. Silvestri [6] for a recent survey). However, although this data has been collected on a large scale—up to 15M queries and 7M sessions in the case of the MSN logs of 2006—it has generally been recorded from a single user population, using either a single website or a single search engine. Since logs have been collected at different times, with different users, and recording different information, direct comparisons are not generally possible.

Given log data from two distinct user populations, with the website and other factors held constant, it would be possible to compare the behaviours of these two populations; and, if there were salient differences, possibly to suggests ways a "one-size-fits-all" tool could be modified to best suit each population's needs. Similarly, if we had comparable log data from the same population interacting with two distinct data sources, it would be possible to compare users' behaviour across different websites or search engines.

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010. Copyright for this article remains with the authors.

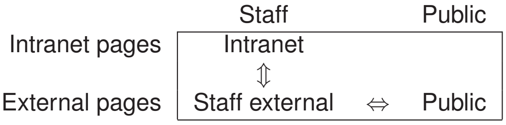


Figure 1: We have three classes of interaction data: from the intranet, from internal users of the external site ("staff external"), and from public users of the external site ("public"). Comparisons are possible across page sets (intranet vs staff external) and across user populations (staff external vs public).

We have explored these ideas with three data sets collected from the internal and external web servers and search engines of a large Australian government agency. Our three sets include two user populations (agency staff and the public) and two sites (one internal to the agency, one open to all). This collection allows two types of comparison (Figure 1):

- comparisons between behaviours of the same users, but on different sets of pages, by looking at staff use of the intranet and of the external webpages (we call this "intranet" and "staff external" data respectively);
- comparisons between behaviours of different users, on the same set of pages, by looking at staff and public uses of the public website (we call this "staff external" and "public" data).

Our experiments considered differences in both these dimensions (Sections 3–5); behaviours which predict when a user may switch from one source to another (Section 6); and morals for the design of web retrieval systems. We begin by describing the logfiles we used.

2 Data

Data was collected from two of the agency's web servers and an associated analytics service. One server was responsible for presenting intranet pages: these pages were only available inside the agency, or over a secure link, and we assume all users of this server were agency staff. The second server was responsible for the agency's public website, and presented pages for users both inside and outside the agency. The servers used different web publishing tools, but the same search engine.

The data covers a one month period (23 October to 22 November 2009), and were collated from search engine logfiles and metrics provided by the "Clicky" tracking service.[1] They represent 498,955 individual sessions and 1,595,180 page views, from 212,255 distinct IP addresses. This is approximately the same size as the logs from the Excite search engine [8] but under 3% the size of the more recent AOL and Microsoft logs.[2] For each session, on either server, we obtained the session start time and duration, the user's IP address (and in many cases their geographical region and network block), and platform details such as operating system and browser. Each session includes a number of "actions"—searches or page views—for which we have time, URL, and referrer data. For searches we also recorded query terms.

Users of the external pages were partitioned into "staff" and "public" on the basis of their IP address: those corresponding to agency domain names, or an agency network block, were assumed to be staff and the remainder were assumed to be the public. This produced 290,493 public sessions (58%); 26,591 staff external sessions (5%); and 181,871 intranet sessions (36%). The majority of staff sessions (87%) were on the intranet.

Note that the intranet search engine maintained logs which were not directly comparable with logs from other sources. We integrated this data by assigning searches to the closest intranet session, with matching user IP address, within 30 minutes of the search. (The 30 minute window is an arbitrary choice. Smaller values however presented similar results.)

Note also that not all of the agency's web presence, internal or external, is captured in our logs—in particular, we only have data from the main, agency-wide, servers. It is not clear how many sessions, on group intranets or groups' public web pages, are not accounted for here. We are confident, however, that we have a large fraction of interactions and those which represent a variety of behaviours and needs.

3 General characteristics

The broad patterns in our data are consistent with those seen in other log files and with commonsense expectations, with a moderate amount of searching and sessions of a few minutes' duration.[3] 61% of public sessions seem to be from Australia, according to hostname and/or geolocation information.

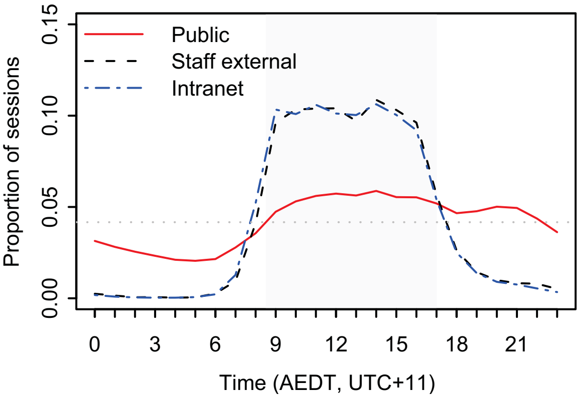


Figure 2: Sessions with each source, by start time.

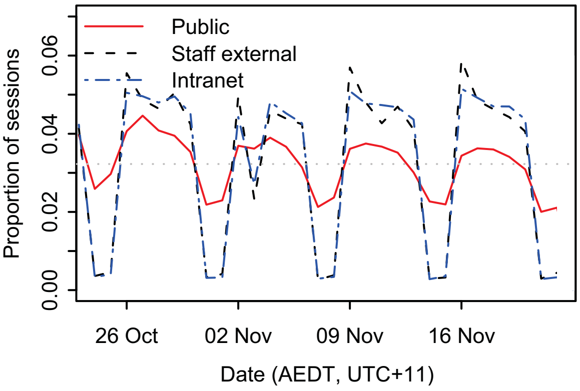


Figure 3: Sessions with each source, by day. Tuesday 3 November 2009 was a public holiday in the ACT and parts of Victoria.

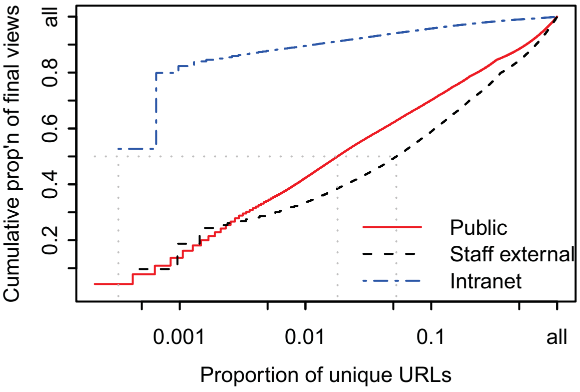


Figure 4: Distribution of final views. Note logarithmic scale on abscissa. From 0.03% to 5.3% of pages, depending on data set, account for 50% of final views.

Figures 2 and 3 summarise the volume of data in our three sets according to session start times. (All times are reported as if in south-east Australia.) There are clear trends: staff use both the internal and external web pages during work hours and on work days. Public usage is more uniform over time, which is consistent with more use from home and more use from outside Australia.

Session entry pages are largely uninteresting, since the distribution is dominated by the public and intranet home pages. Exit pages—the final URL viewed in a

[1] `http://www.getclicky.com/`

[2] The "spring 2006 MSN search data asset" was not publicly released, but see e.g. `http://research.microsoft.com/en-us/events/searchsummit2007/`. Summary statistics are presented in e.g. Zhang and Moffat [9].

[3] We do not have data from any pages viewed prior to the agency's, and in particular we do not have complete data from external search engines which led people to the agency's pages. Search activity in the public and staff external sets will therefore be underestimated, but it is hard to know how much by.

session—are much more likely to have provided the information people require, and patterns here are more interesting.

We considered the distribition of final views across all sessions, except where those sessions were followed by a switch from the internal server to the external, or vice versa (see Section 6). Figure 4 summarises the distribution: reading left to right is an index of URLs, sorted by popularity so that the most popular ending URL is at the far left, and reading bottom to top is the cumulative proportion of final views.

Staff are clearly looking at a small number of intranet pages just before ending a session: the single most popular intranet page constitutes over 50% of final views, as do the two most popular pages from the staff external set. Public interests are more varied and it takes the 85 most popular pages from the public set to make 50% of final views, although this is still only 1.8% of all pages ever viewed. Examining the most popular final URLs shows public interest in the agency's projects and in job advertisements, while staff views are predominantly the web pages of organisational units.

Figure 4 confirms a common skewed pattern: there are some resources used extremely often, and a very long tail. The public data set, in particular, appears to follow a Zipfian distribution but (as is common) this is in fact a poor fit (fitted using the method of Clauset et al. [1], Komolgorov-Smirnov $D = 0.05$, $n = 4717$, $p > 0.05$).

Table 1 summarises a number of other statistics across the three data sets. Superscript [a] indicates significant differences between the public and staff external data (paired t test, $\alpha = 0.05$); that is, differences on the vertical axis of Figure 1 or between the first two columns in the table. Superscript [b] indicates differences between staff external and intranet data (horizontal axis of the figure, columns two and three of the table).

We also examined the proportion of queries (in a subsample) which used proper names; acronyms; and question words. There were only small numbers in each set, and no significant differences.

4 Navigation and effort

The combination of search engine and web server logs allows us to examine browsing and searching behaviour in tandem.

We expect to see some differences between the public and staff external sets, for several reasons: staff will be more familiar with the organisation and language of the agency; staff may be more familiar with the agency websites; and there are likely to be different sorts of tasks for each user group.

In particular, we expect staff to be more successful at navigation, which leads to our first hypothesis:

▷ NAVIGATION: Since staff have better knowledge of the agency's and the websites' structure, they will be more fluent at navigation. Staff sessions will be shorter than those of the public.

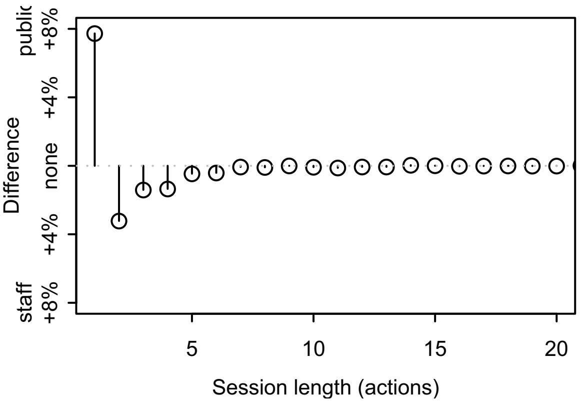


Figure 5: Differences in distributions of session length. Session length reads left to right; differences between public and staff external data sets read up (more common for public) or down (more common for staff).

We also expect that many public users may be browsing for fun, or to scratch an itch, whereas staff are more likely using the website for particular information relevant to their jobs. This leads to two further hypotheses, possibly contrary to the first:

▷ EFFORT #1: Staff are more likely than the public to be highly motivated—especially since they are mostly using the website during the working day, and may be looking for information essential to their work. We expect that staff will not give up if they can't find something immediately.

▷ EFFORT #2: We expect a similar effect as for effort #1, but more pronounced, between the staff external and intranet sets.

The session length statistics in Table 1 seem quite clearly to contradict our navigation hypothesis—rather than being shorter, sessions are instead longer for staff whether measured by actions (a 13% increase) or by time (an 87% increase). This could be explained by the navigation and effort #1 hypotheses together: a larger proportion of staff external sessions could be short, say length 1, while a longer tail (corresponding to extra effort) could drag out the mean.

Figure 5 tests this idea by illustrating the difference in distributions of session length. Session length (in actions) runs left to right; points above the axis are where public sessions of this length are more likely than staff external sessions of this length, and points below are the reverse.

The large spike above the axis for the shortest sessions, and the corresponding mass below the axis at sessions of moderate length, demonstrates that public users are more likely to have a very short session; and staff are more likely to have a longer one. (Less than 10% of sessions in either set are longer than six actions, so while there are similarities in the tail of each distribution these represent only a minority of cases.) Our

	Data set		
	Public	Staff external	Intranet
Scale			
Sessions in set	290,493	26,591	181,871
Actions in set	918,699	95,195	587,344
IP addresses in set	201,452	4,999	10,815
URLs in set	56,790	11,321	23,564
Sessions			
Session length (actions)[ab]	3.2 ± 0.01	3.6 ± 0.05	3.2 ± 0.01
Session length (min:sec)[ab]	3:36 ± 0:01	6:04 ± 0:06	9:00 ± 0:03
Sessions with repeated page views[ab]	24.5%	33.5%	39.1%
Searches			
Sessions with searches[ab]	5.9%	7.9%	7.4%
Queries per session, where > 0[b]	2.5 ± 0.02	2.5 ± 0.08	1.7 ± 0.01
First query length (terms)[a]	2.2 ± 0.01	1.9 ± 0.02	1.9 ± 0.01
Overall query length (terms)[a]	2.3 ± 0.01	1.9 ± 0.01	1.9 ± 0.01
Query groups			
Number of queries per group[ab]	2.1 ± 0.02	1.8 ± 0.06	1.5 ± 0.01
Groups with only one query[ab]	56.9%	68.8%	76.2%
Reformulations			
Groups which added terms[a]	24.8%	14.4%	13.5%
Groups which removed terms[a]	20.1%	12.5%	11.8%
Groups with hypernyms	0.005%	0.002%	0.002%
Groups with hyponyms	0.004%	0.002%	0.002%
Groups with synonyms[a]	0.013%	0.005%	0.006%

Table 1: Summary statistics of the three data sets. [a] indicates significant differences between public and staff external; [b] between staff external and intranet. Shown is mean ± one standard error.

explanation is not supported; we conclude that in fact staff members' sessions are not made shorter by their better organisational knowledge.

The two effort hypotheses, however, are confirmed. Sessions are longer for the staff external set than the public set; they are longer again, in both actions and time, for the intranet set, although changes in layout and language may account for some of this difference.

There is also a significant increase, in both cases, in the number of sessions in which one or more URLs is visited more than once ("repeated page views" in Table 1). This "going in circles" suggests a serious attempt to find some information by navigation—staff are more likely to backtrack and try another path than are the public, and intranet users are more likely again.

We conclude that staff are not in fact more competent at navigation, on either the intranet or the public website, but do expend considerable effort trying to find information important to them. It is likely that the public site is poorly designed for staff use, with information scattered over many pages and forcing extra navigation for staff (but not the public); conventional site analysis or interviews with staff may help confirm or rebut this.

5 Queries, query groups, and reformulations

The combined logs also show differences in search engine use, and in the strategies different groups use to find information.

As before, it seems likely that staff will show some trace of better understanding the agency and the external website. This leads to two likely hypotheses:

▷ QUERIES PER SESSION #1: Staff will be more likely to use useful search terms; and staff will be more familiar with navigational aids and other landmarks in the websites. Therefore, when they do search they will issue fewer queries than the public do.

▷ QUERIES PER SESSION #2: Since the intranet is written for an internal audience, the effects of this familiarity with language and structure should be more pronounced for the intranet set.

The mean number of queries per session is the same for both user groups (Table 1), but the means hide a difference in distribution; staff are more likely to issue a single query in a session, and the public are more likely to issue two to five queries (Figure 6, read the same way as Figure 5). Staff also use slightly fewer terms per query. Overall, there is not strong evidence for the first hypothesis but it is at least plausible.

There is stronger evidence for the second hypothesis. Staff tend to issue fewer queries on the intranet than on the external site (31% fewer overall), and there is again a difference in distributions with intranet sessions around 12% more likely to include a single query and staff external sessions correspondingly more likely to include 2–8.

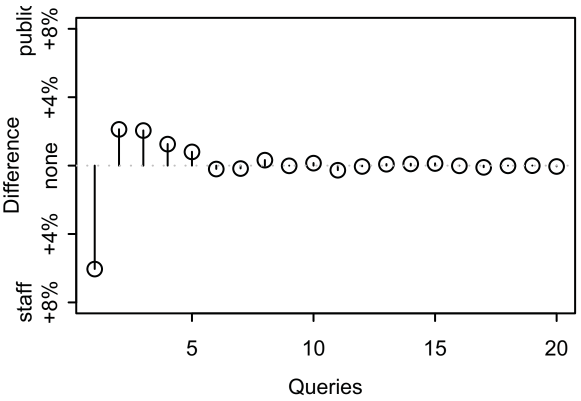


Figure 6: Differences in distributions of query counts. Query counts read left to right; differences between public and staff external data sets read up (more common for public) or down (more common for staff).

Since a session may include any number of searches, possibly for different topics or tasks, we also consider queries as part of "query groups". A "group" attempts to collect queries from the same session which correspond to a single topic. Given a sequence of queries, we created groups in three phases (Figure 7):

1. First, each query was considered in its own group.
2. Second, as in earlier work [9], term similarity was based on the Jaccard similarity between trigrams. Terms were considered "similar" where Jaccard similarity was greater than 0.25.

   Any queries which included similar terms were assumed to be on the same topic, and were grouped together.
3. Finally, any query that was bracketed by queries in some group was assigned to that group. This makes the simplifying assumption that users do not temporarily switch topics in the middle of a sequence.

If we take these query groups into account, there is a stronger effect for both public/staff external and staff external/intranet comparisons (see "query groups" in Table 1). Query groups tend to be shorter, in each case—there is a 15–20% decrease—and significantly more groups only include a single query.

Query reformulations, where a user has repeated a query with modifications, are likely to represent a failed search and are therefore of particular interest. We considered reformulations in each group by counting instances of five relationships between consecutive queries: adding terms; removing terms; including a term which is a hypernym of a term in an earlier query; including a term which is a hyponym of a term in an earlier query; and including a term which is a synonym of a term in an earlier query. Hyper-, hypo-, and synonyms were drawn from WordNet [5] via NLTK [4]. Note that more than one relationship might hold between any two queries: for example,

	Queries			
	butterflies	insects	butterfly flight	bricklaying
1: Each query in its own group	A butterflies	B insects	C butterfly flight	D bricklaying
2: "Butterflies" has 60% character trigram overlap with "butterfly"	A butterflies	B insects	A butterfly flight	D bricklaying
3: "Insects" is between two queries in the same group	A butterflies	A insects	A butterfly flight	D bricklaying

Figure 7: Query grouping example. Four queries in a session are grouped to represent two information needs.

the sequence "vehicle allowance" and "vehicle rules" includes a deletion ("allowance") as well as an addition ("rules").

This data allowed us to test a further hypothesis:

▷ REFORMULATION: Staff will reformulate queries less often than the public do.

This assumes that staff will be more familiar with the agency's language, so will be more likely to choose useful terms early on. (The same effect would also be apparent if staff are so used to the agency's vocabulary that they find it hard to think of rephrasings.)

The reformulation hypotheses is borne out to some extent ("reformulations" in Table 1), and we see fewer reformulations in the staff external set than in the public set although the base rate for hyper-, hypo-, and synonym use is low. (Note that Wordnet's coverage of the terms used in queries was lower than expected, so the rate of hyper-, hypo-, and synonym use is probably higher than reported here.)

There is evidence then that staff are searching less than the public, on the external network, and are reformulating their queries less often. This would normally suggest that their extra knowledge of the agency's structure and language, and their experience with the website, is leading to greater search success. However, the two effort hypotheses were also borne out while the navigation hypotheses was not: and staff tend to have longer sessions, and fewer sessions of length one. Longer sessions with fewer searches do not in fact suggest search success; rather, they suggest that staff are using a different tactic. Recall too that staff were more likely than the public to go in circles, and view a page more than once—a likely sign of navigation, as well as search, failure.

On the basis of search engine log files alone, we may conclude that the smaller number of searches, and smaller number of repeated searches, mean that staff are generally successful; that the search engine is doing a good job. With the integrated logs, the opposite becomes clear: there are definite signs of increased effort (sessions are longer) and of staff abandoning the search engine in favour of navigation (fewer queries are issued). That navigation also seems unsuccessful a lot of the time (repeated views are high). Web server logs clearly add something important to the search engine logs and having only one or the other would tell only half the story.

6 Switching sources

Staff have access to both the internal and external web pages. It is possible therefore for staff to look for information from one source which is in fact available from the other: for example, it is possible to spend some time looking for street addresses in the intranet although they are generally published to the public web. Our last set of questions investigate occasions when staff switch sources—that is, when they stop using one data source in favour of another.

Session identifiers in the logs are not comparable across sources, so switches were determined by IP address: a switch was counted when the same IP address appears in both the staff external and intranet sets (in either order) with a gap of less than thirty minutes. Smaller gaps made minimal difference.

Table 2 summarises statistics for sessions which did switch, and sessions which did not switch, but could have (i.e. staff external or intranet sessions), in the same way as Table 1.

▷ LITTLE SWITCHING: As before, we expect that agency staff—who use the intranet and the public site on a daily basis—know where to look for different classes of information, at least at the site level. We expect to see little switching.

There were 10,117 instances of switching from 208,462 staff external or intranet sessions. This is a fairly small proportion (only 4.9%), but it does represent a significant effect. Sessions which end with a switch tend to involve more effort (mean 120% more actions) and be much longer (mean 112% more time). If the 17 minutes 31 seconds spent before switching sources to look somewhere else represents

	Sessions ending with	
	Switch	No switch
Scale		
Sessions in set	10,117	198,345
Actions in set	69,061	614,607
IP addresses in set	3,541	11,223
Sessions		
Session length (actions)$^{\mathrm{c}}$	6.8 ± 0.11	3.1 ± 0.01
Session length (min:sec)$^{\mathrm{c}}$	17:31 ± 0:17	8:15 ± 0:02
Sessions with repeated page views$^{\mathrm{c}}$	57.1%	39.8%
Searches		
Sessions with searches$^{\mathrm{c}}$	20.1%	6.8%
Queries per session, where $> 0^{\mathrm{c}}$	2.4 ± 0.06	1.8 ± 0.02
First query length (terms)$^{\mathrm{c}}$	2.0 ± 0.02	1.8 ± 0.01
Overall query length (terms)$^{\mathrm{c}}$	2.0 ± 0.02	1.9 ± 0.01
Query groups		
Number of queries per group$^{\mathrm{c}}$	1.8 ± 0.04	1.5 ± 0.01
Groups with only one query$^{\mathrm{c}}$	69.8%	76.0%
Reformulations		
Groups which added terms$^{\mathrm{c}}$	16.9%	13.1%
Groups which removed terms$^{\mathrm{c}}$	14.9%	11.4%
Groups with hypernyms	0.003%	0.001%
Groups with hyponyms	0.003%	0.001%
Groups with synonyms	0.006%	0.006%

Table 2: Summary statistics of sessions which do or do not end with a switch of sources. $^{\mathrm{c}}$ indicates significant differences. Shown is mean ± one standard error.

wasted time, then this is over 140 hours per day of lost time—equivalent to eighteen full-time staff. This may be an underestimate since the logs used here only capture session switches, not sessions abandoned entirely in favour of external search engines or other methods entirely.

The sessions which end with a switch do show evidence of users struggling with both search and navigation. As well as being dramatically longer, switching sessions are 43% more likely to include repeated page views. They are three times more likely to involve querying, and when the search engines are used there are more queries, queries are slightly longer (although this is a small effect), and query groups are 26% more likely to include a second or subsequent query.

As well as trying harder, users who are about to switch sources show some signs of adopting different strategies. Query reformulation is also somewhat more common in switching sessions than otherwise, although still less common than in the public set.

If it were possible to predict which sessions will end in a switch, based on browsing and searching data like that in Table 2, it should be possible to offer extra help in these sessions. For example, if there is reason to believe a staff member is looking in the wrong place then we could expand the search engine's scope to include other sources. Depending on what sort of intervention is proposed, such a predictor will most likely need high precision but could sacrifice some recall.

Unfortunately whole-of-session features such as user IP address, session length, number of repeated page views, number of searches, and number of query groups are not promising in this regard. Neither J48 trees built with WEKA [2] nor SVMs built with SVMlight [3] have been able to predict a final switch with greater than 65% accuracy.

Rather than aggregate over the actions in a session, a more promising technique is to examine each separately and consider *sequences* of actions as cues that users might be struggling. As a first attempt, we sampled 400 sessions from the logs—200 which switched and 200 which were candidates, but did not—and coded each action as a search, a page view, or a repeated page view (i.e. a view of a page already seen in the same session). Each session was represented by all its subsequences of length two or more. For example, a session with the sequence search, view, view, re-view—"svvr"—would be represented by the subsequences { svvr, svv, vvr, sv, vv, vr }.

There were over 9000 subsequences in our sample which only occurred in switching sessions, and 243 which occurred in two or more switching sessions and no non-switching sessions. These included obvious subsequences such as "sssss"—five searches in a row, with no other page views—and long runs of repeated views. 33 of our 400 sampled sessions included one or more of these 243 subsequences, so using their presence as a cue it would be possible to detect about 16% of switching sessions before the point of the switch, and with very high accuracy. By including other subsequences it would be possible to increase recall at the expense of precision.

These subsequences of actions could be used in combination with other techniques: for example, it may be possible to improve predictions by using the presence of certain subsequences as a feature in an SVM classifier. Analysis of maximal repeating patterns [7] may also offer insights, and we intend investigating these further.

7 Conclusions and further work

Combined logs from two servers, two search engines, and two user populations have allowed us to contrast users' behaviours across information sources and across populations. While the overall patterns are similar to other logs, there are differences in searching and browsing behaviour in both comparisons.

Although staff presumably are more familiar with the agency's organisation, language, and website they spend more effort finding information online. This may be because they are more motivated; but certainly their familiarity does not seem to translate to greater success at navigation or searching. This is likely due to poor site design, but further analysis or interviews are needed to better understand staff behaviour and to point to particular areas needing improvement.

With the combined logs we can also observe some sessions which end with a switch in source, from the intranet to the public site or vice versa, and these represent a considerable amount of staff time. Early indications are that we may be able to spot these sessions on the fly, with high precision, to offer extra help. Further work will pursue this idea.

Many of these observations would not be possible given only a search engine log, or only a web server log. The combination is useful and we hope to expand future logfile work to include both sources where possible.

Acknowledgements We are grateful for help and input from Tom Rowlands, and from the agency's information technology group. We also thank the anonymous reviewers for their useful comments.

References

[1] Aaron Clauset, Cosma R Shalizi and M E J Newman. Power-law distributions in empirical data. *SIAM Review*, Volume 51, Number 4, pages 661–703, 2009. arXiv:0706.1062v2.

[2] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann and Ian H. Witten. The WEKA data mining software: An update. *SIGKDD Explorations*, Volume 11, Number 1, pages 10–18, 2009.

[3] Thorsten Joachims. Making large-scale SVM learning practical. In Bernhard Schölkopf, Christopher J.C. Burges and Alexander J. Smola (editors), *Advances in kernel methods: Support vector learning*, pages 169–184. MIT Press, Cambridge, Massachusetts, 1999.

[4] Edward Loper and Steven Bird. NLTK: The natural language toolkit. In *Proc. ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pages 63–70, Philadelphia, July 2002. Association for Computational Linguistics. arXiv:cs/0205028v1.

[5] George A Miller. Wordnet: a lexical database for English. *Comm. ACM*, Volume 38, Number 11, pages 39–41, 1995.

[6] Fabrizio Silverstri. *Mining query logs: Turning search usage data into knowledge*, Volume 4 of *Foundations and Trends in Information Retrieval*. Now Publishers, Delft, 2010.

[7] Antonio C Siochi and Roger W Ehrich. Computer analysis of user interfaces based on repetition in transcripts of user sessions. *Trans. Info. Systems*, Volume 9, Number 4, pages 309–335, 1991.

[8] Amanda Spink, Bernard J Jansen, Dietmar Wolfram and Tefco Saracevic. From e-sex to e-commerce: Web search changes. *Computer*, Volume 35, Number 3, pages 107–109, 2002.

[9] Yuye Zhang and Alistair Moffat. Some observations on user search behaviour. In *Proc. ADCS*, pages 1–8, 2006.

Relatively Relevant: Assessor Shift in Document Judgements

Mark Sanderson *Falk Scholer*
School of Computer Science and Information Technology
RMIT University
Melbourne, Australia
{*mark.sanderson,falk.scholer*}@*rmit.edu.au*

Andrew Turpin
Department of Computer Science and Software Engineering
The University of Melbourne
Melbourne, Australia
aturpin@unimelb.edu.au

Abstract *The evaluation of information retrieval systems relies on relevance judgements – human assessments of whether a document is relevant to a specified search request. In the past, it was demonstrated that test collection assessors disagree with each other to some extent on the relevance of documents and can be inconsistent in themselves. This paper describes a series of investigations on assessor consistency, which demonstrate that the inconsistency of an assessor varies over time. We show that when documents are presented to assessors in a relevance independent order, documents judged as relevant appear to cluster. Examining pairs of documents in a sequence ordered by time-of-judgement, we find that relevance assessors judge highly similar document pairs more consistently when the pairs are seen soon after each other; the consistency reduces when the pairs are judged further apart. We contend that our analysis shows that changes are not due to random error, but instead reflect a* relevance shift*, whereby the assessor's conception of what constitutes a relevant document changes over time. Studying types of relevance judgement we find that the shift in judgements is greatest between highly and partially relevant documents. We also examine the impact of this inconsistency on how retrieval runs are ranked relative to each other and find that there appears to be a noticeable effect on such rankings.*

Keywords Information retrieval evaluation, Cranfield approach, Relevance judgements, TREC.

1 Introduction

Relevance judgements are a key component of test collections, which were defined in the "Cranfield methodology", the principal paradigm by which information retrieval systems are evaluated [5]. Soon after Cleverdon and colleagues proposed the use of test collections, objections were raised which focussed on the anticipated inconsistency of the assessors who would form such judgements. Such was the force of these criticisms that early IR test collection creators, Cleverdon and Salton, both conducted studies to understand the importance of assessor variability [5, 9]. They found that although assessors differed in their view about which documents were relevant, the way in which systems were ranked based on the different judgements was generally unaffected. As the size of test collections grew, the studies initiated by Salton and Cleverdon were repeated in subsequent decades coming to largely the same conclusions [21, 24].

While there has been a strong focus on examining the impact of assessor consistency, there has been less study on the nature of the inconsistency. Salton, Cleverdon and later Sanderson [10], showed that assessors tended to agree on the relevance of top ranked documents; they were less consistent on the lower ranked. Chen and Karger suggested that differences in assessment were linked to different interpretations of what a topic meant [4]. Bernstein and Zobel, examining the gov2 collection, showed that individual assessors did make errors in judgement. They found a noticeable number of documents in a test collection that were textually almost identical but had been judged differently by the same assessor [1].

One of the potential sources of error in relevance judgements is the queries, which are often poorly specified. This can present a challenge to assessors on determining exactly what is and is not relevant. In the absence of a detailed specification, we hypothesise that assessors will look back at the documents they judged earlier to contrast with the document they currently have to assess. If assessors have a large number of documents to judge for a particular topic, earlier decisions may be forgotten and the documents used as a comparison to make relative judgements will change, leading to a

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

shift in assessment criteria over time. In other areas where humans are used for assessment of documents, such as marking coursework, shifts in assessment are well understood; methods to control it such as detailed grade related criteria or even score standardisation tests are used to minimize this effect [12, 17].

To avoid biases in relevance assessments related to the rankings of retrieval systems, judges are presented with documents in the order in which they appear in the collection; in effect, an arbitrary order. Therefore, the data from the TREC collections that is stored in the relevance files (called *qrels*) is stored in the same order in which the documents were judged by the assessors (as explained by Harman [8], and confirmed in a personal communication with TREC staff). This data therefore provides a valuable resource from which it is possible to study relevance shift. This paper describes a series of preliminary experiments that were conducted on TREC data to better understand if relevance shift exists in test collections; if it does, what the nature of the shift is; and what impact any shift has on the way that retrieval systems are ranked by the test collection.

The rest of this paper is structured as follows: in Section 2 we present the background and related work on the evaluation of information retrieval systems, and previous studies into relevance assessments. Section 3 discusses relevance shift and how such a phenomenon might be identified in relevance judgements. We then present a series of experiments to test our hypotheses. Discussion and possible directions for future work are then given in Section 4.

2 Background and Related Work

Information retrieval systems are evaluated to determine how effectively they are able to help users fulfill information needs. The most widely-used approach for the evaluation of information retrieval systems is through the use of test collections. This approach, known as the Cranfield methodology, is a simulation of the search process [5]. A number of test *queries* are run across a fixed *collection* of documents using the retrieval system that is to be evaluated. For each query, the system generates a ranked answer list, with documents ordered by their estimated likelihood of being relevant to the search request. For each answer item that is returned, a human assessor then makes a *relevance judgement*, indicating whether the document is relevant to the search request, or not.

Based the answer lists of an IR system and the human relevance judgements, a range of performance metrics can be calculated to quantify the effectiveness of a retrieval system. These are generally based on precision (the number of relevant documents that were retrieved as a proportion of the total number of documents retrieved), recall (the number of relevant documents retrieved as a proportion of the total number of available relevant documents), or both. Mean average precision (MAP) is perhaps the most widely-reported performance metric. For a single query, average precision is defined as the mean of the precision scores obtained at each point where a relevant document is retrieved in a ranked answer list; MAP is then the mean of the average precision scores across a set of search topics. MAP has been shown to be a stable evaluation metric, and reflects both the precision and the recall of a retrieval system [2].

The Cranfield paradigm is used in IR evaluation campaigns including the Text REtrieval Conference (TREC) [22]. In TREC it is common practice for relevance judgements to be made by paid *assessors*, typically retired information analysts, who are asked to behave as if the provided search tasks are real information needs. The judging instructions stipulate that a document is to be judged as relevant if it contains any information that would be used in writing a report about the search topic under consideration. Answers are shown to assessors sorted by document number, to avoid potential bias from the ranking position of items in system answer lists; as a result, the judging instructions ask assessors to consider each document independently of all others (that is, there is no concept of redundant information). Moreover, to promote consistency of judgements, the documents to be judged for each topic are assigned to a single assessor [8].

Relevance is a vital concept for the evaluation of information retrieval systems, since it is the ability of such systems to provide useful answer documents – that is, documents that help a user to solve an information need – that determines their overall utility. While different levels of relevance have been proposed in the information science literature, in Cranfield-based evaluation of information retrieval systems it is typical to focus on *topical* relevance, where the focus is on the relation between a document and the topic under consideration [11]. In this operationalisation of relevance, user context is abstracted out, with the intention of allowing greater consistency of judgements. Nevertheless, many factors that can impact on the variability of relevance judgements have been identified, including requirements, statement variables, document variables, judgment conditions, judgment scales, and personal factors [6].

Despite limiting relevance assessments to be topical, analysis of judgements has shown surprisingly low levels of inter-rater agreement. A comparison of the relevance judgements of three different TREC assessors for TREC-4 topics showed an overlap of 0.42–0.49, while overlap with re-assessments of TREC-6 judgements by university students gave an overlap of only 0.33 [20]. In another study, reassessment of 40 TREC newswire topics from TREC-6 to TREC-8 showed that around 64% of documents were judged differently [16]. However, despite these relatively low levels of agreement when assessing individual documents, relative *system orderings* obtained when competing IR systems are evaluated based on different judgement sets were found to be generally stable, with a Kendall's tau correlation between system orderings of around 0.9 [20]. This level of correlation is widely taken to be representative as a threshold level of disagreement that should be expected when evaluating system orderings, due to noise from variation in relevance judgements.

While the design of IR evaluation campaigns seeks to limit inconsistencies in relevance assessments, it is widely acknowledged in the cognitive science community that people's choices and decisions are not consistent, and are sometimes not even rational [7, 13, 18]. In this paper we examine evidence for changes in the criteria applied by assessors when making relevance judgements.

3 Identifying Relevance Shift

To study shift in relevance criteria during the judging of a sequence of documents for a single topic, it is necessary to know the order in which relevance judgements were made by assessors. As described above, given a search topic t, the relevance data (*qrels*) are defined as a list of documents $\{d_1, ..., d_n\}$, where n is the number of documents judged for t. The assessors are shown the documents in the order 1 to n. While it is possible that assessors may stray from this order while judging, this form of behaviour is not thought to be common. We also specify R^t_d to be a *relevance judgement* detailing the relationship between t and d. For the TREC `wt10g` and `gov2` collections,

$$R^t_d = \begin{cases} 2, & \text{if } d \text{ is } \textit{highly relevant} \text{ for } t \\ 1, & \text{if } d \text{ is } \textit{relevant} \text{ for } t \\ 0, & \text{if } d \text{ is } \textit{not relevant} \text{ for } t. \end{cases}$$

For the TREC 7 and 8 *adhoc* collections only relevant and not relevant levels were specified (R^t_d is 0 or 1). Therefore, given a pair of documents d_i and d_j in the *qrels* list, we assume that the time period between the two documents being assessed is proportional to the *distance* $|i - j|$ between the documents in the list. With that assumption established, a number of tests were conducted to see if an assessor's view of relevance changed across the judgements. They are now described.

3.1 Distance between pairs of relevant and non-relevant documents

The first investigation attempted to check for clustering of relevant documents in the *qrels* list by comparing the distance between randomly selected pairs of relevant documents and similarly selected pairs of documents judged not relevant. If relevant documents are distributed uniformly throughout the *qrels*, then the distances should be equal, on average. If an assessor's conception of relevance shifted over time while judging the *qrels* list, then we would expect clustering of relevant documents in the *qrels* list. For example, similar documents found close to each other in the list may all be judged relevant, but an equally relevant document seen later may be judged irrelevant if the assessor shifts their relevance criteria.

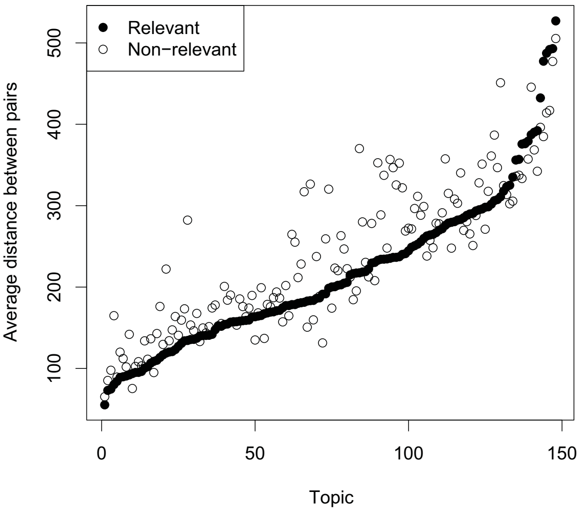


Figure 1: Average distance between pairs of relevant and non-relevant documents for the Terabyte track topics on the `gov2` collection. The x-axis has been sorted by increasing distance between relevant pairs.

It is important that other sources of relevance clustering are not present in the *qrels*, hence the TREC collections based on newspaper data could not be used. In such collections the document ids were related to the temporal order of the articles in the newspapers, so relevant documents would be expected to cluster around news stories reported intensely over a limited time period. It was decided to use *qrels* for topics 701 to 850 of the `gov2` web collection of ".gov" web pages. Here the document ids relate to the order in which the US government web sites were crawled.

This set of *qrels* still has some (known) features that may cause clustering of relevance judgements. First, due to the crawling process, documents from the same web site – which might be similarly relevant to a particular topic – were likely to be close to each other in the *qrels* list. Second, a large set of PDF documents were crawled towards the end of the gathering of the `gov2` collection, which means that most HTML documents are found in the first part of the *qrels* while most PDF documents are at the end.

Therefore, for each topic in `gov2`, we first examined the distance between randomly selected pairs of relevant documents that weren't from the same domain and were both an HTML file; second we tested for pairs that were both a PDF file. The result of this test is shown in Figure 1 where for each topic, the average distance between relevant and non-relevant pairs of HTML documents is shown. As can be seen, across a large number of topics the distance between relevant pairs of documents is smaller than the distance between non-relevant. Averaged across the topics,

	$R^t_{d_i} \neq R^t_{d_k}$	$R^t_{d_i} = R^t_{d_k}$
gov2	248	165
wt10g	455	207
t7t8	89	63

Table 1: Comparison of distance between pairs of similar documents in the *qrels* of three TREC test collections. Columns show inconsistent and consistent judgements respectively.

the distance was found to be statistically significant ($p < 0.01$). Here the significance test used was a randomization test, where multiple random partitions of the *qrels* were formed in the same per-topic proportion of relevant and non-relevant documents. When measuring distances between pairs of items randomly drawn from a set, the average distance is influenced by the size of the set. The distance between pairs drawn from small sets is likely to be shorter than for large sets. However, the difference in distance between the relevant and non-relevant was significantly larger than the difference found in the random sets. For more details on the randomization test's use in information retrieval, see Smucker [15].

The test was repeated for pairs of relevant and non-relevant documents where both were PDF files, and a similar statistically significant difference between the distances was found. Overall, the distance between randomly selected pairs of relevant documents in the gov2 *qrels* is smaller than the distance between similarly selected pairs of non-relevant documents. This implies that there are clusters of relevant documents in the *qrels*. However, it is possible that this is occurring because the relevance files were arranged in an order that is not independent of relevance. Therefore, the next experiment was conducted.

3.2 Distance between pairs of highly similar documents for which judgements are consistent or inconsistent

It has been noted in several past papers [1, 3] that within the TREC *qrels* sets, there are pairs of documents that are very similar to each other, but which assessors judged differently; there are also similar pairs that assessors judged consistently. Both types of document pairs occur in sufficient quantity for them to be studied in the context of this work. We hypothesised that there were two possible explanations for the inconsistent assessor behaviour: simple error, or relevance shift. If assessors were occasionally making mistakes, then the average distance between pairs that were judged the same or differently would be equal. On the other hand, if the inconsistency was due to relevance shift, then one would expect assessors to be consistent for pairs of documents seen soon after each other, and inconsistent for pairs seen far apart.

	$R^t_{d_i} \neq R^t_{d_k}$	$R^t_{d_i} = R^t_{d_k}$
gov2	237	165
wt10g	510	207

Table 2: Comparison of distance between pairs of similar documents in the *qrels* of two TREC test collections. Columns show inconsistent and consistent judgements respectively. Here, only inconsistent pairs for *not relevant* and *partially relevant* are counted.

	$R^t_{d_i} \neq R^t_{d_k}$	$R^t_{d_i} = R^t_{d_k}$
gov2	265	165
wt10g	330	207

Table 3: Comparison of distance between pairs of similar documents in the *qrels* of two TREC test collections. Columns show inconsistent and consistent judgements respectively. Here, only inconsistent pairs for *partially relevant* and *highly relevant* are counted.

	$R^t_{d_i} \neq R^t_{d_k}$	$R^t_{d_i} = R^t_{d_k}$
gov2	236	165
wt10g	221	207

Table 4: Comparison of distance between pairs of similar documents in the *qrels* of two TREC test collections. Columns show inconsistent and consistent judgements respectively. Here, only inconsistent pairs for *not relevant* and *highly relevant* are counted.

Here for a series of TREC test collections, the text of relevant documents from the *qrels* were in turn used as a query to search for other documents that were very similar to the "query document" and that had also been assessed for relevance. Similarity was calculated using the cosine measure across all content terms of judged documents (stemming and stopping were not applied). Any documents with a similarity of 0.9 or more were retained. Next, the distance in the *qrels* between the document pairs was measured. For this experiment, both web and newspaper based TREC collections were used. The concerns about clusters of relevant documents in the *qrels* of newspaper data was not a problem here, since if the inconsistent judgements were due to simple random assessor error there would be no difference in average distances.

The results are summarised in Table 1. As can be seen across all three tested collections, the mean distance between documents judged consistently (where the relevance of the document pair was the same) was substantially less than the distance between those judged inconsistently. We take this to indicate strong evidence that an assessor's view on what is and is not relevant changes over time.

As the wt10g and gov2 collections had two levels of relevance, there were three different types of inconsistent judgement:

qrels position	312	…	582	…	712
Doc	d_i	…	d_j	…	d_k

Figure 2: For a single topic, given a document d_i that is judged as relevant, d_j and d_k $(j < k)$ are the furthest away documents that have cosine similarity with d_i of ≥ 0.9. The distance between d_j and d_k (130 in this instance) divided by the distance between d_i and d_k (400) is reported in Figure 3 for all relevant d_is.

- 0 and 1 - not relevant and partially relevant
- 1 and 2 - partially relevant and highly relevant
- 0 and 2 - not relevant and highly relevant

The differences for these three were tabulated in Tables 2, 3 and 4. The tables show that the distance between inconsistent judgements is smallest between the not relevant and highly relevant classes. This is particularly true for `wt10g`, where the difference in distance between consistent and inconsistent judgements is very small, indicating little or no relevance shift. For the `wt10g` collection the largest distance is between relevant and partially relevant documents, and for `gov2` it is between partially and highly relevant documents. In both cases, the more marginal relevant judgements appear to be more prone to shift than less marginal judgements.

3.2.1 Considering intervening documents

One possible reason why similar documents that are far apart in the *qrels* are judged inconsistently while closer similar documents are not is that there is less chance for the assessor to "forget" the relevance criteria used on the first document in the pair by the time they come to the second. If the two documents are close in the *qrels*, then only a small number of other documents have been judged, and so perhaps the assessor can maintain a consistent model of their relevance criteria. A further aid to maintaining a consistent relevance criteria could be that there are other documents between the pair that are also similar, and so they serve as a "reminder" to the judge of their criteria.

As a first attempt at measuring such an effect in the newspaper and web datasets we examined the similarity scores for documents between pairs, computing the distance of the closest document to the second of the pair. Specifically, for a pair of highly similar documents d_i and d_k, let d_j be the closest highly-similar document to d_k that lies between the pair in the *qrels*. Figure 2 shows a schematic of the approach we adopted.

For each relevant document d_i in the *qrels*, d_j and d_k are located, and the ratio of the distance between d_j and d_k to the distance between d_i and d_k is computed. The resulting triple is classed as *Same* if $R^t_{d_k} > 0$ (that is, both d_i and d_k are relevant), and *Different* if $R^t_{d_k} = 0$ (d_i is relevant while d_k is irrelevant).

`wt10g`	Same	1.7%	
	Different	1.5%	$p < 10^{-14}$
`gov2`	Same	2.7%	
	Different	1.3%	$p < 0.006$

Table 5: Proportion of documents in the interval between d_i and d_k in *qrels* order that have a cosine similarity score ≥ 0.9 to d_i. The final column shows the p value from a t-test of the row and the row above.

Figure 3 shows the ratios for the two collections. A ratio value of 1 indicates that there are no documents similar to d_i between d_i and d_k (that is, $d_j = d_i$), whereas a ratio close to 0 indicates that d_j was judged just prior to the judgement of d_k. Hence, we hypothesise that if the ratio is close to one, it is more likely the pair will be Different as there has been no similar "reminder" document before the judgment of d_k. It is clear in Figure 3 that the ratio is higher for Different pairs than for Same pairs in our two data sets. This difference is statistically significant (t-test, $p < 0.0001$ for `wt10g`, $p < 0.003$ for `gov2`), giving evidence that reminder documents are present for Same pairs, and not for Different pairs.

Also, simply counting the proportion of similar documents that occur in between d_i and d_k shows significantly more for Same pairs than for Different pairs, as shown in Table 5, again providing evidence that reminder documents may be present, and contribute to consistent relevance judgements, while a lack of such reminder documents leads to relevance shift.

3.3 Ordering retrieval systems based on subsets of relevance judgements

The criteria used to judge the relevance of documents may change over time, despite the best efforts of assessors. As we have argued, if this is the case, we would expect subtle differences between the documents that were judged as being relevant early in a series of relevance judgements, compared with those that are judged as being relevant later.

A further way to investigate such a shift is to consider the impact on performance scores when systems are evaluated using judgements from early in the *qrels*, compared with evaluating systems using judgements that were made later.

The effectiveness of information retrieval systems is commonly evaluated by calculating system performance measures such as MAP. Such scores are meaningful for a particular collection of documents, and set of test topics, but are not comparable across different collections or sets of search requests [23]. IR experiments therefore typically focus on *relative* effectiveness scores, for example, comparing the MAP scores of a "baseline" system with those of a "new" retrieval approach, with both systems running a common set of search topics over the same collection. Similarly, where a common set of queries is run over a single collec-

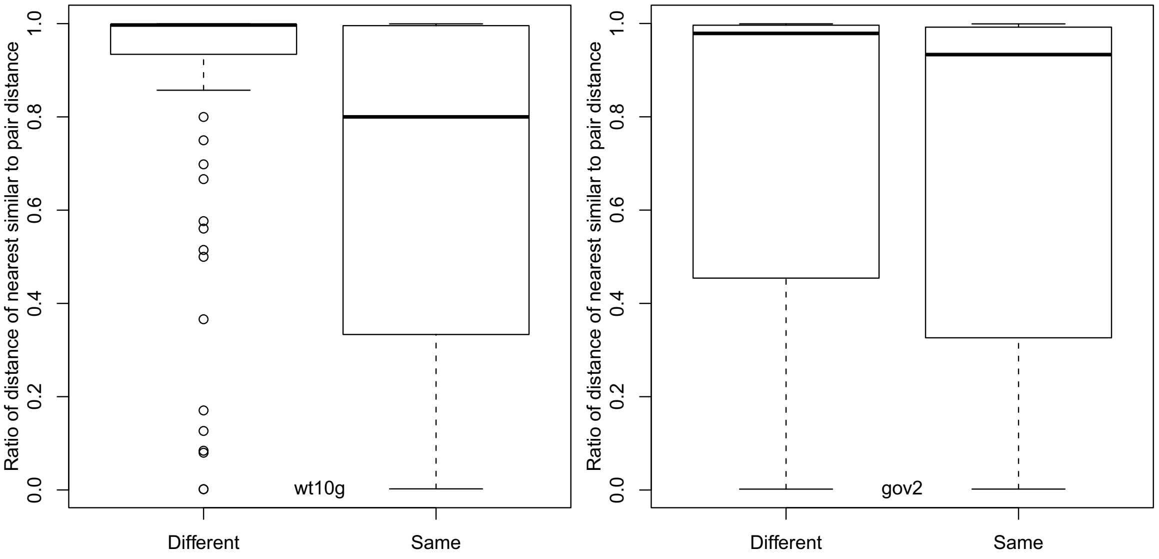


Figure 3: Ratio of the distance between the closest similar doc to the end of a pair (d_j and d_k) to the distance between the first and last of the pair (d_i and d_k) in *qrels* order.

tion with multiple systems, for example in evaluation campaigns such as TREC, it is the overall *ordering* of retrieval system performance that is of interest.

When a set of retrieval systems is evaluated using two different sets of relevance judgements, the relative performance of the systems may differ. The impact that changing relevance judgements has on the relative system scores can be measured using Kendall's τ (tau). Tau measures the extent to which two rankings agree, and is equivalent to the number of pairwise swaps that are required to obtain one ranking from the other [14]. The value of tau can range from +1 (perfect agreement) to -1 (perfect disagreement).

The assumption of relevance shift would suggest that systematic differences can arise between relevance judgements that were made at different stages of the judging process. In other words, we would expect there to be a difference in the relative ordering of systems when evaluated using relevance judgements from different parts of a *qrels* set (a tau score of less than 1). Moreover, if such a difference was systematic and reflected relevance shift, then the difference observed when using relevance judgements that were made early in the assessment process, compared to those that were made late in the process, should be greater than the difference that would be observed when randomly partitioning the set of relevance judgements (that is, not taking judgement order and possible relevance shift into account).

We test this hypothesis using data from the 2006 Terabyte track, which includes relevance judgements for topics 701–850 on the gov2 collection. 80 runs were submitted to the track, representing different retrieval systems (or configurations of retrieval systems). Because runs that are submitted to TREC are based on experimental systems, they may contain bugs or have other problems. It is therefore common practice to discard the 25% lowest performing runs [21]. We discard the 20 runs with the lowest MAP scores based on the official full set of relevance judgements for our analysis below.

The original, ordered, relevance judgements are first partitioned into two halves, selecting the first 50% of relevant documents for each topic to be the "early" set, and the second 50% to be the "late" set. The 60 system runs are then evaluated based on their MAP scores when using the two partitioned relevance judgements. Kendall's tau between the two obtained system orderings is 0.493, showing substantial disagreement between the two orderings.

As a comparison, we randomly partition the relevant documents for each topic into two halves, and similarly evaluate the 60 runs using the split relevance judgements. Figure 4 shows the tau values obtained for 50 random partitionings of the relevance file. The mean tau score of the random runs is 0.752; moreover, the tau score from the split ordered relevance judgements is substantially lower than even the smallest value obtained for a random split, providing evidence for the presence of systematic variation in the ordered relevance judgements, and the impact that relevance shift may have on the evaluation of information retrieval systems.

4 Discussion

People's beliefs, opinions and criteria for making decisions change over time. In this paper we have carried out an initial set of experiments to investigate whether there is post-hoc evidence for the existence of relevance

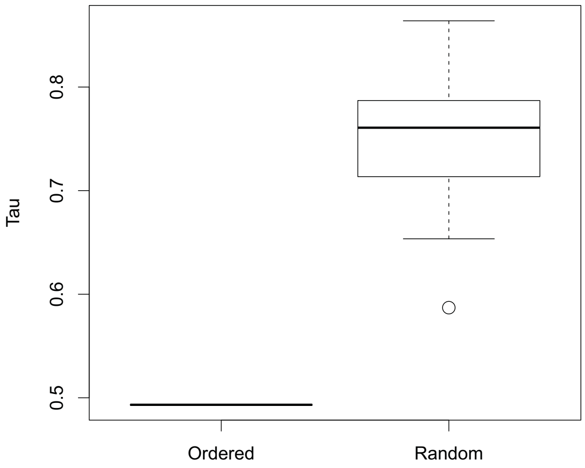


Figure 4: Kendall's tau correlation between system orderings when evaluated with split relevance judgements. using the original ordering versus random partitioning.

shift during judging of official relevance judgements made by TREC assessors.

Our results demonstrate that documents that are judged as being relevant tend to be clustered together as assessments are made, suggesting that coherent judging criteria are maintained most strongly in bursts.

One possible reason for this might be the mode in which TREC topics are developed and subsequently assessed. In particular, topics are chosen so that they will have relevant answer documents in the collection being used. Potential topics that do not have "enough" relevant documents are discarded, as are topics that have "too many" relevant documents [19]. Most times the assessment of relevance for topics is conducted by people involved in the topic development process, and so they have an a priori mental model of the number of relevant documents that should appear in the final *qrels*. During judging, then, if they have just deemed a run of documents to be relevant, they may alter their criteria to be more strict so that the total number of relevant documents does not get too high. Conversely, if there has been a run of non-relevant documents, they might loosen their criteria. Naturally any gross alterations in criteria could involve going back and re-judging documents subject to the new criteria, but perhaps this does not happen, or happens erroneously.

For pairs of highly similar documents, our results demonstrated that inconsistencies in judgements are not due to random error, but are again affected by the distance between the documents, with a greater distance leading to a higher likelihood of an inconsistent assessment. Furthermore, the presence of highly similar documents as assessments are being made impacts positively on the consistency of relevance judgements, suggesting that these help an assessor to maintain a consistent set of judging criteria. Finally, the presence of relevance shift appears to have a systematic impact on system assessments, with assessments based on judgements from early or late in a *qrels* showing much greater variation than when the judgements are partitioned randomly.

Taken as a whole, the results offer compelling evidence for the existence of relevance shift when large numbers of document judgements are being made for a search topic. However, our results so far have been from empirical analysis of relevance files, and have abstracted away from investigating the details of the documents themselves.

It should be remembered that in the experiments using pairs of highly similar documents, the presence of relevance shift is not restricted to just those pairs; relevance shift could well be occurring in judgements of documents elsewhere in the *qrels*. Therefore, in future work, we intend to carry out a user study where explicit assessments of pairs of documents from different parts of a *qrels* are made. We will also track which parts of documents are considered as being relevant, allowing more fine-grained analysis. User data will be key in further differentiating between cases where inconsistencies in judgements are made due to simple errors in attention, compared with shifts in relevance criteria. We also intend to further investigate the impact of the search topic specifications on relevance shift, using measures of clarity and readability to analyse the query statements.

Acknowledgements We thank Ian Soboroff for helpful discussions about the TREC assessment process and relevance judgements.

References

[1] Y. Bernstein and J. Zobel. Redundant documents and search effectiveness. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 736–743. ACM, 2005.

[2] Chris Buckley and Ellen M. Voorhees. Retrieval system evaluation. In Ellen M. Voorhees and Donna K. Harman (editors), *TREC: experiment and evaluation in information retrieval*. MIT Press, 2005.

[3] S. Bttcher, C. L. A. Clarke, P. C. K. Yeung and I. Soboroff. Reliable information retrieval evaluation with incomplete and biased judgements. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 63–70. ACM Press New York, NY, USA, 2007.

[4] H. Chen and D. R. Karger. Less is more: probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 429–436. ACM New York, NY, USA, 2006.

[5] Cyril Cleverdon. The cranfield tests on index language devices. *Aslib Proceedings*, Volume 19, pages 173–192, 1967. (Reprinted in K. Sparck Jones and P. Willett, editors. *Readings in Information Retrieval*. Morgan Kaufmann Publishers Inc., 1997).

[6] Carlos Cuadra and Robert Katter. The relevance of relevance assessment. In *Proceedings of the American Documentation Institute*, Volume 4, pages 95–99, 1967.

[7] Benedetto de Martino, Dharshan Kumaran, Ben Seymour and Raymond J. Dolan. Frames, biases and rational decision-making in the human brain. *Science*, Volume 313, pages 684–687, 2006.

[8] Donna K. Harman. The TREC test collection. In Ellen M. Voorhees and Donna K. Harman (editors), *TREC: experiment and evaluation in information retrieval*. MIT Press, 2005.

[9] M. Lesk and G. Salton. Relevance assessments and retrieval system evaluation. *Information storage and retrieval*, Volume 4, Number 4, pages 343–359, 1968.

[10] M. Sanderson. Accurate user directed summarization from existing tools. In *Proceedings of the seventh international conference on Information and knowledge management*, pages 45–51. ACM New York, NY, USA, 1998.

[11] T. Saracevic. Relevance: A review of the literature and a framework for thinking on the notion in information science. Part II: nature and manifestations of relevance. *Journal of the American Society for Information Science and Technology*, Volume 58, Number 13, pages 1915–1933, 2007.

[12] M. N. K. Saunders and S. M. Davis. The use of assessment criteria to ensure consistency of marking: some implications for good practice. *Quality Assurance in Education*, Volume 6, Number 3, pages 162–171, 1998.

[13] James Shanteau. Psychological characteristics and strategies of expert decision makers. *Acta Psychologica*, Volume 68, Number 1-3, pages 203 – 215, 1988.

[14] David Sheskin. *Handbook of parametric and nonparametric statistical procedures*. CRC Press, 1997.

[15] M. D. Smucker, J. Allan and B. Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 623–632. ACM New York, NY, USA, 2007.

[16] Eero Sormunen. Liberal relevance criteria of TREC – counting on negligible documents? In *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 324–330, Tampere, Finland, 2002.

[17] I. Suto, R. Nádas and J. Bell. Who should mark what? a study of factors affecting marking accuracy in a biology examination. *Research Papers in Education*, pages 1–31, 2009.

[18] Amos Tversky and Daniel Kahneman. The framing of decisions and the psychology of choice. *Science*, Volume 211, pages 453–458, 1981.

[19] E. M. Voorhees and D. Harman. Overview of the fifth text retrieval conference (TREC-5). In *The Fifth Text Retrieval Conference (TREC-5), Gaithersburg, MD, USA*, NIST Special Publication, Gaithersburg, MD, USA, 1996. Department of Commerce, National Institute of Standards and Technology.

[20] Ellen M. Voorhees. Variations in relevance judgements and the measurement of retrieval effectiveness. *Information Processing and Management*, Volume 36, Number 5, pages 697–716, 2000.

[21] Ellen M. Voorhees and Chris Buckley. The effect of topic set size on retrieval experiment error. In *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 316–323, Tampere, Finland, 2002.

[22] Ellen M. Voorhees and Donna K. Harman. *TREC: experiment and evaluation in information retrieval*. MIT Press, 2005.

[23] William Webber, Alistair Moffat and Justin Zobel. Statistical power in retrieval experimentation. In *Proceedings of the ACM CIKM International Conference on Information and Knowledge Management*, pages 571–580, Napa Valley, California, USA, 2008.

[24] Justin Zobel. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 307–314, Melbourne, Australia, 1998.

The Impact of Users' Cognitive Style on Their Navigational Behaviors in Web Searching

Khamsum Kinley

Faculty of Science and Technology
Queensland University of Technology
Brisbane QLD 4001 Australia

k.kinleyd@gmail.com

Dian Tjondronegoro

Faculty of Science and Technology
Queensland University of Technology
Brisbane QLD 4001 Australia

dian@qut.edu.au

Abstract *User-Web interactions have emerged as an important area of research in the field of information science. In this study, we investigate the effects of users' cognitive styles on their Web navigational styles and information processing strategies. We report results from the analyses of 594 minutes recorded Web search sessions of 18 participants engaged in 54 scenario-based search tasks. We use questionnaires, cognitive style test, Web session logs and think-aloud as the data collection instruments. We classify users' cognitive styles as verbalisers and imagers based on Riding's (1991) Cognitive Style Analysis test. Two classifications of navigational styles and three categories of information processing strategies are identified. Our study findings show that there exist relationships between users' cognitive style, and their navigational styles and information processing strategies. Verbal users seem to display sporadic navigational styles, and adopt a scanning strategy to understand the content of the search result page, while imagery users follow a structured navigational style and reading approach. We develop a matrix and a model that depicts the relationships between users' cognitive styles, and their navigational style and information processing strategies. We discuss how the findings from this study could help search engine designers to provide an adaptive navigation support to users.*

Keywords Web Searching, Navigational Style, Information Processing Strategy, User Cognitive Style.

1 Introduction

User-Web interactions have emerged as an important area of research in the field of information science. As new technology emerges, different information systems have been developed for improving Web searching and information retrieval. However, Web users often report difficulties in Web searching. Search effectiveness may be affected by many factors specific to topics and task, of which a user's cognitive style - an individual's preferred and habitual approach to organizing, perceiving, remembering, and representing information [26], have been found to be influential in affecting searching [9, 17]. This has motivated further investigations and more researchers are now exploring the Web search behavior from a user's perspective. Earlier, the information systems and the intermediaries, who manage them were concerned about information use from the system's perspective; they have focused on designing questions, searching strategies or queries that best match the system's representation of texts rather than responding to users' problems when retrieving the information [18].

In order to investigate users' issues and problems in retrieving information from the Web, it is imperative to understand users' Web navigations, information searching and retrieving processes, and cognitive factors, such as users' cognitive styles that influence these processes. This study first investigates users' cognitive styles, Web search navigations and information processing strategies, and then reports on the relationships between these components. We define Web searching as "all-users activities during the logging on/logging off period" on Web or information system [28], and cognitive style as an individual's preferred and habitual approach to organize and represent information [26]. There is no such thing as bad or good cognitive style, but an individual with a certain cognitive style tends to find certain tasks easier than others.

2 Related Studies

The study of how users navigate the Web, and the impact of their user characteristics, such as cognitive style, on the Web search behavior is a significant contemporary topic. Different authors refer to cognitive style with different terms, such as field-dependent/independent [30], holists-serialist [21], and wholist-analytic/verbal-imagery [25]. Field-dependence-independence describes the degree to which an individual's perception or comprehension of information is affected by the surrounding fields [30]. Riding and Cheema [25] grouped the cognitive

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

dimensions into two principal cognitive dimensions: the Wholist-Analytic and the Verbal-Imagery style dimensions. The *Wholist-Analytic* (WA) dimension of cognitive style describes the habitual way in which people think about, view and structure information in wholes or parts. This affects the way they learn and organize information. The *Verbal-Imagery* (VI) dimension of cognitive style describes an individual's tendency to process information either in verbal or verbal mode of representation and thinking. It refers to ways in which an individual would represent knowledge in either words (verbal) or mental pictures (images).

A number of tools are available to assess cognitive styles [23, 24, 29]. Riding's [24] *Cognitive Style Analysis* (CSA) test is a computer presented test to measure WA and VI dimensions of cognitive styles [25] by means of a ratio. The CSA comprises of three sub-tests. The first part assesses the VI dimension by presenting series of statements on one at a time to be judged true or false. Half of the statements contain information about conceptual categories, while the other half describes the appearance of items. The computer then records the response time to each of the statements and calculates the VI ratio. A low ratio (below 0.98) corresponds to a verbaliser, a high ratio (1.09 and above) to an imager, while the intermediate position being described as 'Bimodal'[24]. However, many researchers [6, 7] tend to use a dichotomous classification by grouping into two groups: Verbaliser and Imager. The second and third sub-tests assess the WA dimension of cognitive styles by presenting series of geometrical figures and the individual is required to judge the figures. In this paper we use Riding's CSA test to classify participants into verbal or imagery cognitive styles.

Most search engines today provide multiple navigation tools to allow users to structure their navigation strategies with multiple approaches. For example, Google provide different search features and tools, such as maps, image, and video; users can use these tools to search information. However, studies have reported users getting lost or disoriented while navigating on the Web. Chen and Macredie [4] reported users confronting "disorientation problem", "lost in hyperspace", and "mismatching" while navigating on the Web. They also reported a user's preference, such as his or her cognitive style, having significant effects on his or her navigation. Field-Dependent students preferred guided navigation (linear), while Field-Independent learners preferred freedom of navigation.

Kim [14] investigated how users' emotion control and search tasks interact and influence the Web search behavior and performance among experienced Web users. The study findings indicated that users tended to use more navigation tools in a general search task that required them to find a few pieces of information on a broad topic than in a specific task that required locating one specific piece of information that was known to exist on the Web.

From a user study exploring the relationships between Web users' searching behavior and their cognitive style, Kinley, Tjondronegoro and Partridge [17] presented a conceptual model of Web searching and cognitive styles. The model presented based on the preliminary findings, revealed relationships between different stages of Web searching and cognitive factors. Among the cognitive factors, the cognitive style of a user was found to have a greater impact. As the authors reported, the study results and the model presented are in its "infancy" as the findings were based on a small scale population sample.

2.1 Limitations of the Current Studies

The studies discussed in the previous section provide valuable insights into cognitive styles and Web searching research, in particular Web navigations. They are the bases on which this study is founded upon. However, their findings on Web navigations were based on a low level variables, such as either the number of clicks on navigational buttons [14, 15], or the counts of visits to web pages [10], which do not implicitly represent a user's navigation patterns. There exist limited empirical studies that have looked into the relationships between users' cognitive styles, and their navigational and information processing strategies. In fact, there is no empirical study conducted on investigating the effects of users' cognitive styles on their information processing strategies. To the best of our knowledge, this study is the first work exploring this area of investigation.

In this study we first look into how users search the Web and then investigate the effects of their cognitive styles on their Web navigations and information processing strategies. Investigation into users' navigational style and information processing strategies, and their cognitive styles, will provide rich data about user-Web interactions.

3 Research Aims and Questions

Users' navigational style and information processing strategies are important elements of Web search behavior because they are the path towards successful Web searching. They are like tools that can add extra leverage in searching and retrieving the required information.

Studies show that a user possesses unique characteristics [25, 26, 30]. Among these characteristics, cognitive style is one of the most important user factors that affects Web searching and information search performance [7, 14, 19].

This study aims to investigate the effects of users' cognitive styles on their Web navigations and information processing strategies. The findings in this study will help search engine designers to provide an adaptive navigation support to users. The fundamental research question underpinning this research is:

What are the relationships between users' cognitive styles, and their Web search navigations and information processing strategies?

4 Methodology

4.1 Study Participants

A total of 18 volunteers (8 male and 10 female), comprising of 8 postgraduate research students, 2 academics and 8 professional staff from the Queensland University of Technology participated in the study. The participants' age was between 20 years and 56 years old. They regularly engage and search the Web for information in the course of their academic, personal or administrative activities.

4.2 Search Tasks

We developed three search tasks, outlined in Table 1, based on Borlund and Ingwersen's [2] concept of "simulated work task situation" or scenarios. The search tasks were designed to ensure that these tasks are as close as possible to the real world situations. The simulated work task situation provides each searcher with the context, which ensures "a degree of freedom" to react in relation to his or her interpretation of the given situation [2]. This approach has been used by several researchers in information seeking studies [examples include: 1, 13].

The search tasks were also designed with different levels of difficulty and complexity, and a diverse area of topics. Task 1 presented the least complexity, which required using basic searching skills. Task 2 was more complex and required a higher level of search experience than for task 1. Task 3 was more complex compared to task 2 and required participants to use a more advanced level of search terms and presented relatively more abstract scenarios compared to task 1 and task 2. Although many studies [examples include: 6, 10, 13] show task type as an influential factor in Web searching, it is not a controlled variable in this study. We aim to investigate the effects of task complexity and difficulty on Web search behavior in future.

4.3 Data Collection

Riding's [24] *CSA* test was used to classify participants into verbal or imagery cognitive styles. The CSA test indicates the position of an individual on the VI fundamental style dimensions by means of a ratio, which describes an individual's tendency to process information either in words (verbal) or mental pictures and thinking (images). We use a dichotomous classification; a low ratio (below 1.03) on the VI scale corresponds to a verbaliser, while a high ratio (1.03 and above) to an imager. Although there has been a few studies questioning its reliability and validity [20, 22], the CSA test was chosen in this research because of the following points, (1) the CSA test is relatively new compared to Embedded Figure Test [29] or Verbaliser-Visualiser Questionnaire [23]; (2) the CSA test has been shown to have good reliability and validity, and a good number of studies have used the test [examples includes: 6, 7, 8]; and (3) CSA test is a computer administered test which often makes it more attractive to participants and also makes data collection easier for researchers.

Task 1: You have recently moved to Austin, Texas, USA and would like to know the relevant laws passed by the Texas state government regarding child safety while travelling in vehicles. Identify three such rules.

Task 2: You, with your two friends, are planning a trek for one week in Solukhumbu in Nepal. The trekking will occur next month. You are told that tourists trekking in the place may get high-altitude illness. You decide that you should know more about the place, and symptoms, seriousness and preventions of high-altitude sickness.

Task 3: There has been talk of the Bermuda Triangle mystery for the last three decades or so. You are curious about the mystery and want to know more about it. So, you want to search any incidents, people's views and any other relevant information (literature, images and videos) about it.

Table 1: Search Tasks

Participants' cognitive thinking was collected through a think-aloud method. Think-aloud method is used for investigating a user's cognitive process, which requires the participant to verbalize as he or she performs specified search tasks. We investigate users' interactions and their navigational styles with the Web search systems by investigating their Web search sessions. We use a monitoring program to record Web search sessions and think-aloud protocols.

4.4 Procedure

Participants' demographic information were collected using a pre-experiment questionnaire. Following the cognitive style test, the participants were then invited to participate in the Web searching experiment; they were assigned three sets of search tasks, outlined in Table 1. Although the participants were never stopped while performing their search tasks, it was recommended that they spend between 10 and 15 minutes on each search task.

Participants were asked to talk aloud while they were performing the search tasks. They received the following instructions.

> You are required to verbalize orally your thoughts, motivations, actions, and reasons while conducting a Web search. This will enable the researcher to understand your cognitive thinking.

Their Web interactions, including think-aloud and Web search logs, were captured using a monitoring program.

4.5 Data Analysis

The captured user-Web interactions for each participant were played and replayed several times to create participant observation memos with search logs, session length, and think-aloud stamps. The total session length was 594 minutes. Important search behavior then emerged from the search logs were coded for qualitative analysis using elements of content analysis [11, 27] and protocol analysis [5] within a constructivist grounded theory approach [3].

5 Results

The participants' demographic information indicated that they had a minimum of 3 years Web search experience and were skilled with at least basic searching skills. Although participants' demographic information contributed significantly to this study, participants were not differentiated by their demographic data as it is not a controlled variable in this study. In this study we focus on participants' cognitive styles and its impact on how they navigate the Web and process information.

5.1 Cognitive Style

Based on the VI ratio, we classify participants into two groups: *verbalisers* and *imagers*. Participants scoring below 1.03 on the VI scale were classified as *verbalisers* and those scoring 1.03 or above as *imagers*. Out of 18, 10 participants were classified as having a verbal cognitive style while, 8 participants were imagery users.

5.2 Web Search Patterns

Based on the findings that emerged from the qualitative analysis, two types of Web search patterns were identified for the study: *Web Navigational Styles* (NS), based on how users navigate during Web searching, and *Information Processing Strategies* (IPS), based on how they view and process search results or retrieved result pages.

Web Navigational Styles

To investigate the participants' navigational style, we classify navigational styles into two categories: sporadic and structured navigations that bear some similarities to those suggested in previous studies [10, 12].

Sporadic navigational style refers to those behaviors in which users performed an unstructured navigation during Web searching. They visited numerous links and pages, switching between browser tabs and windows, and were thereby characterized by a shorter duration between any two consecutive nodes. They opened many pages simultaneously and quickly scanned each of these pages. They tended to navigate back and forward more often, Users formulated queries, read first few lines, navigated back to the search result page, and then reformulated the query; they seemed to repeat the same procedure again.

Users with sporadic navigational styles also took some time to decide on search terms to be used, and links and pages to be visited or clicked. They tended to view only the first few search result pages and seldom clicked on the 'Next' button of the search results page. They also tended to visit the homepage more frequently and used the 'back' button more often, which is an indication that they felt uncertain about their searching.
Users of this kind were found to be unorganized. Palmquist and Kim [19] relate frequent usage of embedded links to a 'passive' way of navigation and use of Home button as an indication of 'getting lost' that is, stopping whatever they have been doing and starting over again. As per Palmquist and Kim's interpretation, this indicated that sporadic users get lost more frequently than the rest.

Structured navigation style is associated with relatively a lesser use of links of the site visited, longer periods between any two nodes and low homepage use. Users preferred to use multi windows to navigate and manage information; they used separate windows to manage links/pages of similar topics. Participant 7 pointed out:

> "I like to have a few windows opened at the same time and look for different subjects. So in one window will be looking for hotels, food, etc and other one will look up activities."

Users seemed to feel confident about their searching performance. They focused on a fewer pages and read carefully in detail. They spent relatively a longer duration on each page they visited. They performed one thing at a time, spent adequate time on a single task and navigated cautiously from one page/search to another.

Information Processing Strategies

Information processing strategies refer to approaches adopted by users to view, select and process information during Web searching. Based on the study findings and our previous study results [16], three information processing strategies are identified: Scanning, Reading, and Mixed strategies.

Scanning refers to examining hastily, where a user makes a sweeping search for a piece of information. Users formulated and reformulated their queries more often, clicked several links, opened numerous result pages and scanned them quickly. The time span between any two consecutive nodes was relatively shorter. Users were also found switching between subject topics, and between browser tabs and windows. For instance, the first thing Participant 1 did with the results from his first query was to quickly scan the search result descriptions, and then he formulated his query without opening or reading the result pages. Users were found scanning result pages for general information without a clearly defined goal.

Reading refers to a comprehensive viewing, examining and understanding the information on a page. Users visited relatively a lesser number of pages in a given duration and spent relatively a longer time on a page. Users were found reading pages in details and spent enough time to understand the content of a page. They often opened links and pages in the same window, which indicated that they preferred to read a single page and accomplish one task at a given time. For an example, Participant 14 was cautious about what she was searching for. She opened one page at a time and based on the information retrieved with the preceding query, she reformulated her query carefully. For instance, having found the general information, i.e. a map on Solukhumbu, she then searched for other information on accommodation.

A *mixed* strategy of information processing involves both scanning and reading. During the experiment, it was observed that some participants adopted both scanning and reading in parallel to process information. At a certain point of their searching and examination, users started and stopped scanning, and then switched to reading. Few users were found scanning and reading result pages either at the same time in multiple windows or at different stages of their searching. Initially, Participant 8 formulated and reformulated his queries several times. Most of the time the user followed repetitive search behavior- formulating a query, scanning the search result descriptions, and reformulating the query without opening any retrieved result pages. However, at a certain point he was found reading a result page in detail for more than 3 minutes. There were a handful of users who processed information by both scanning and reading.

5.3 Impact of Cognitive Styles

Previous section has demonstrated that Web users use different navigational styles and processing strategies to search and access information. Next we investigate the relationships between verbal and imagery users, and their Web search behavior in relation to the two Web patterns identified earlier. We report our findings on how users with different cognitive styles, i.e. verbalisers and imagers, navigate the Web and process information.

Web Navigational Styles

Verbalisers: We observed that in general, verbal users seemed to exhibit sporadic navigational styles. They tended to open many links and pages, and used 'back' and 'homepage' buttons more frequently. They were found to be impatient with their search as they frequently scanned the result pages, which seemed to make them confused. They also reported more dissatisfaction with their search results (Participant 2 and Participant 9) and some users displayed frustration with the search (Participant 3 and Participant 9). While searching a map on Solukhumbu in Nepal, Participant 9 pointed out, "I should be looking for Nepal map. I am not very happy with that [retrieved page]".

Verbal users tended to use multiple navigational features, such as clicks, back button, home button, and history. In general they seemed to employ trial and error strategies to find the needed information.

Imagers: In general, imagery users appeared to follow structured navigational strategies while searching information on the Web. They concentrated on a single page and visited relatively a lesser links but they ensured to read them in details. Users seemed to be more organized with their Web searching and followed step-by-step navigations. For instance, Participant 14, who is an imager, followed systematic Web navigations.

> "This person trekked to Sagarmatha National Park. I don't know what it [the park] is and I have no idea about this place. I need to go back and have a better understanding of Solukhumbu, geographical part of it and understand map of it." (Participant 14)

Having found the map of Nepal with Solukumbu district and having a better understanding of Solukhumbu, the user then searched for other information.

> "Let me have a look on the Map of Nepal. This one [map] has the map of Solukhumbu. Sagarmatha National Park [map] in Solukhumbu has blue area [shaded with blue color] showing me where Solukhumbu is. That is very good. So I have now a better understanding of Solukhumbu. Solukhumbu district is a part of Sagarmatha zone. I have now a better understanding of what that area is. Next, I need to search where to stay". (Participant 14)

Information Processing Strategies

Verbalisers: Although there were a few vebalisers who adopted reading approaches, in general they seemed to prefer scanning to process information. They scanned through the search result descriptions and result pages to see if they contain the required information or not. For instance, Participant 1 formulated and reformulated his queries more often, opened several result pages and scanned them quickly. This behavior was repeated several times throughout the entire searching.

Imagers: On the contrary, imagery users tended to prefer reading; they were found reading result pages in detail and spent an adequate amount of time to understand the content of the pages. They visited relatively a lesser number of pages in a given duration and spent relatively a longer time on a page. Participant 4 with imagery cognitive style opened the first result page in the same window and spent more than 3 minutes reading the page in detail. Throughout the search tasks, the participant was found reading carefully and spent sufficient time (approximately 3 minutes) on each result page she opened. In fact, she spent more than 10 minutes on the first two queries. It was also observed that most of the time she opened the result page in the same window, which indicated that she preferred to read one page at a time.

6 Discussion and Implications

We reported results from the analyses of 594 minutes recorded Web search sessions of 18 participants engaged in 54 scenario-based search tasks. The captured Web search interactions and think-aloud exercises provided excellent data into users Web search behavior; users adopted different navigational styles and information processing strategies. Our study found significant associations between users' cognitive styles, and their navigational style and information processing strategies. Users with sporadic navigational styles tended to navigate the Web in a non-linear mode. They tended to visit their homepage more frequently and used the 'back' button more often. They were unable to reconstruct their navigation paths and therefore were prone to get stuck. On the contrary, structured navigators followed a linear navigation. They followed defined steps and retrieved information more effectively than others. They focused on a fewer pages, spent adequate time and cautiously navigated from one page/search to another.

We also observed that as participants progress with their searching, they tended to try various alternatives on a trial and error basis. Participant 16 initially displayed a structured navigational style, but towards the end of the task 3, his navigational style changed to sporadic style, where he clicked the 'Next' button several times without a proper examination of the search results. In fact, he navigated till page 6 of the Google image search result, which is worth noting because this was the first of such kind observed in our experiment. As he navigated hastily while performing search task 3, the participant only scanned the result pages, whereas he spent a sufficient time on reading the pages while performing task 1 and task 2.

To give a clear overview of our study findings, we developed a matrix and a model that depicts the relationships between users' cognitive style, and their navigational style and information processing strategies. Figure 1 summarizes the attributes of the two classifications of navigational styles and three categories of information processing strategies that emerged during the data analyses. The dashed line between the imagery user and sporadic navigations indicates relatively a fewer number of imagery users displaying sporadic navigations, which needs to be reconfirmed in future studies. Table 2 illustrates a

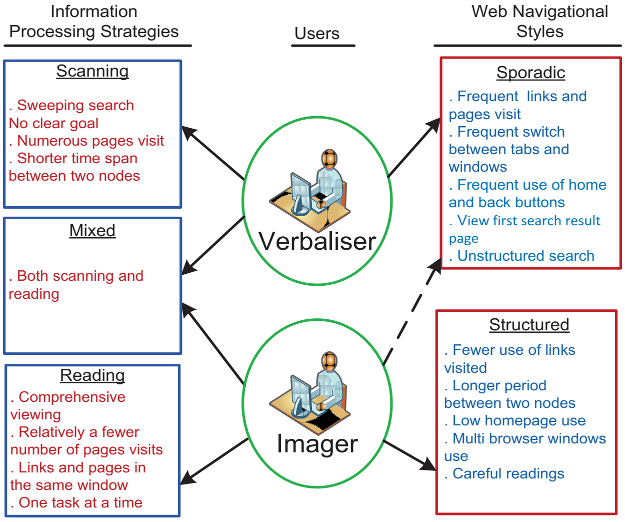

Information Processing Strategies
Users
Web Navigational Styles
Scanning
. Sweeping search
No clear goal
. Numerous pages visit
. Shorter time span between two nodes
Mixed
. Both scanning and reading
Reading
. Comprehensive viewing
. Relatively a fewer number of pages visits
. Links and pages in the same window
. One task at a time
Verbaliser
Imager
Sporadic
. Frequent links and pages visit
. Frequent switch between tabs and windows
. Frequent use of home and back buttons
. View first search result page
. Unstructured search
Structured
. Fewer use of links visited
. Longer period between two nodes
. Low homepage use
. Multi browser windows use
. Careful readings

Figure 1: Relationships between users' cognitive style and their information processing strategies and navigational styles

UserID	CS		NS		IPS		
	V	I	SP	ST	S	R	M
1	○			○	○		
2	○		○		○		
3	○		○		○		
5	○			○		○	
7	○			○		○	
8	○		○				○
9	○		○		○		
10	○		○				○
11	○		○		○		
13	○		○				○
4		○		○		○	
6		○		○			○
12		○	○		○		
14		○		○		○	
15		○		○		○	
16		○		○			○
17		○		○		○	
18		○		○		○	

Note: Cognitive Style (**CS**); Navigational Styles (**NS**); Information Processing Strategies (**IPS**); Verbaliser (**V**); Imager (**I**); Sporadic (**SP**); Structured (**ST**); Scanning (**S**); Reading (**R**); Mixed (**M**).

Table 2: Cognitive Style-Navigational Style – Information Processing Matrix

matrix of cognitive style, navigational style and information processing strategies.

This study has demonstrated that a user's cognitive style plays an important role in Web searching and navigations. Cognitive style affects users' Web search navigations and information processing strategies. The next question we should consider is:

How can we provide adaptive navigation and effective information retrieval?

Search engine designers need to be aware that users differ their cognitive styles, and that a user with a certain cognitive style tends to navigate in a structured manner, while others follow sporadic navigations. Some users find certain search tasks easier, while others experience difficulties. Web search engines can utilize our findings to provide a better search assistance according to users' cognitive styles and their navigational styles. For instance, systems can provide effective browsing tools with an interactive user interface, such as webpage embedded with interactive navigation buttons and links, to users with sporadic navigational styles. Web pages can have advance bookmark features, which enable users to keep a track of their searching and navigations. Similarly, search engine may store in-depth subject contents with diverse topics, so that users with structured navigational styles can explore extra information related to their search task and information need.

7 Limitations

Although this study has successfully illustrated valuable findings into users' cognitive styles, and their Web navigations and information processing strategies, it has some limitations. The study data were collected from a total of 18 end-users participants. Small sampling of participants prevents advance statistical analysis of the data, thus, prevents from illustrating statistical correlation significance. The grounded qualitative analyses would have been boasted had it been supported with statistical methods, such as correlation analysis, factor analysis, and analysis of variance (ANOVA).

As illustrated in Figure 1 by a dashed line between imagery users and sporadic navigations, while most of the imagery users follow structured navigational styles, few of them tended to follow sporadic navigations. There may be other factors, such as query formulation strategies and task complexity, which might have influenced the user's Web navigational style and information processing strategies. Although many studies [6, 10, 13] show task type as an influential factor in Web searching, in this study we have not considered the effects of the task complexity and difficulty on Web navigations and information processing strategies. Further intensive investigations, involving both qualitative analysis and quantitative statistical analyses with a larger sample population, are needed to reconfirm the findings presented.

8 Conclusion and Future work

The findings reported in this paper provide valuable insights into the Web search behavior of users with different cognitive styles. Users' Web search behavior, in particular, their navigational styles and information processing strategies, appear to be affected by their cognitive styles. Verbal users seem to navigate in a non-linear mode, while, imagery users take a more linear approach. Table 2 and Figure 1 depict the Web search patterns and the relationships between users' cognitive styles, and their Web navigations and information processing strategies.

We aim to conduct similar research in the future with a larger sample population not only to reconfirm the results presented in this study but, also to investigate how users with different cognitive styles formulate their queries, what kinds of results they click, and how they deal with task complexity and its effect on their search. This will contribute to a better understanding of Web search behavior from a user's perspective, which will help search engine designers to provide users a better Web search support.

References

[1] P. Borlund. The IIR Evaluation Model: A Framework for Evaluation of Interactive Information Retrieval Systems. *Information Research,* Volume 8, Number 3, pages 8-3, 2003.

[2] P. Borlund and P. Ingwersen. The development of a method for the evaluation of interactive information retrieval systems. *Journal of documentation,* Volume 53, Number 3, pages 225-250, 1997.

[3] K. Charmaz, *Constructing Grounded Theory: A Practical Guide Through Qualitative Analysis*, Sage Publications Ltd, 2006.

[4] S. Chen and R. Macredie. Cognitive styles and hypermedia navigation: Development of a learning model. *Journal of the American Society for Information Science and Technology,* Volume 53, Number 1, pages 3-15, 2002.

[5] K. Ericsson and H. Simon, *Protocol Analysis : Verbal Reports as Data*, MIT Press Cambridge, Mass, 1993.

[6] N. Ford, B. Eaglestone, A. Madden, and M. Whittle. Web Searching by the "general public": An Individual Differences Perspective. *Journal of Documentation,* Volume 65, Number 4, pages 632-667, 2009.

[7] N. Ford, D. Miller, and N. Moss. The Role of Individual Differences in Internet Searching: An Empirical Study. *Journal of the American Society for Information Science and Technology,* Volume 52, Number 12, pages 1049–1066, 2001.

[8] E. Frias-Martinez, S. Y. Chen, and X. Liu. Investigation of behavior and perception of digital library users: A cognitive style perspective. *International Journal of Information Management,* Volume 28, Number 5, pages 355-365, 2008.

[9] G. Gorrell, B. Eaglestone, N. Ford, P. Holdridge, and A. Madden. Towards "metacognitively aware" IR Systems: An Initial User Study. *Journal of documentation,* Volume 65, Number 3, pages 446-469, 2009.

[10] J. Gwizdka and I. Spence. Implicit measures of lostness and success in web navigation. *Interacting with Computers,* Volume 19, Number 3, pages 357-369, 2007.

[11] H. Julien. A Content Analysis of the Recent Information Needs and Uses Literature. *Library & Information Science Research,* Volume 18, Number 1, pages 53-65, 1996.

[12] I. Juvina and H. Oostendorp. Individual Differences and Behavioral Aspects Involved in Modeling Web Navigation. *Lecture Notes in Computer Science,* Volume 3196, pages 77-95, 2004.

[13] J. Kim. Describing and predicting information-seeking behavior on the Web. *Journal of the American Society for Information Science and Technology,* Volume 60, Number 4, pages 679-693, 2009.

[14] K.-S. Kim. Effects of emotion control and task on Web searching behavior. *Information Processing & Management,* Volume 44, pages 373-385, 2008.

[15] K.-S. Kim and B. Allen. Cognitive and Task Influences on Web Searching Behavior. *Journal of the American Society for Information Science and Technology,* Volume 53, Number 2, pages 109-119, 2002.

[16] K. Kinley and D. Tjondronegoro. User-Web Interactions: How Wholistic/ Analytic Web Users Search the Web? In *the 22nd Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design - Interaction - Participation*, pages 344-347, Brisbane, Australia, 2010.

[17] K. Kinley, D. Tjondronegoro, and H. Partridge. Web Searching Interaction Model based on User Cognitive Styles. In *the 22nd Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design - Interaction - Participation*, pages 340-343, Brisbane, Australia, 2010.

[18] C. Kuhlthau. Inside the Search Process: Information Seeking from the User's Perspective. *Journal of the American Society for Information Science,* Volume 42, Number 5, pages 361-371, 1991.

[19] R. Palmquist and K. Kim. Cognitive Style and On-line Database Search Experience as Predictors of Web Search Performance. *Journal of the American Society for Information Science,* Volume 51, Number 6, pages 558–566, 2000.

[20] A. Parkinson, A. Mullally, and J. Redmond. Test–retest Reliability of Riding's Cognitive Styles Analysis Test. *Personality and individual differences,* Volume 37, Number 6, pages 1273-1278, 2004.

[21] G. Pask. Styles and Strategies of Learning. *British Journal of Educational Psychology,* Volume 46, Number 1, pages 128-148, 1976.

[22] E. R. Peterson, I. J. Deary, and E. J. Austin. Celebrating a common finding: Riding's CSA test is unreliable. *Personality and individual differences,* Volume 43, Number 8, pages 2309-2312, 2007.

[23] A. Richardson. Verbalizer-Visualizer: A Cognitive Style Dimension. *Journal of Mental Imagery,* Volume 1, Number 1, pages 109-126, 1977.

[24] R. Riding, *Cognitive Styles Analysis*, Learning and Training Technology, Birmingham, 1991.

[25] R. Riding and I. Cheema. Cognitive Styles—An Overview and Integration. *Educational Psychology,* Volume 11, Number 3, pages 193-215, 1991.

[26] R. Riding and S. Rayner, *Cognitive Styles and Learning Strategies: Understanding Style Differences in Learning and Behaviour*, David Fulton, London, UK, 1998.

[27] L. Schamber. Time-line Interviews and Inductive Content Analysis: Their Effectiveness for Exploring Cognitive Behaviors. *Journal of the American Society for Information Science,* Volume 51, Number 8, pages 734–744, 2000.

[28] A. Spink, H. C. Ozmutlu, and S. Ozmutlu. Multitasking Information Seeking and Searching Processes. *Journal of the American Society for Information Science and Technology,* Volume 53, Number 8, pages 639-652, 2002.

[29] L. Thurstone, *A Factorial Study of Perception*, University of Chicago Press Chicago, 1944.

[30] H. Witkin, C. Moore, D. Goodenough, and P. Cox. Field-dependent and field-independent cognitive styles and their educational implications. *Review of educational research,* Volume 47, Number 1, pages 1-64, 1977.

Criteria that have an effect on users while making image relevance judgements

Rahayu A Hamid
School of Computer Science and IT
RMIT University
Victoria 3001 Australia
rahayu.ahamid@student.rmit.edu.au

James A. Thom
School of Computer Science and IT
RMIT University
Victoria 3001 Australia
james.thom@rmit.edu.au

Abstract *This paper reports the result of an exploratory user study investigating criteria that are important to users when judging relevance while performing an image search. Data was collected from 12 participants using questionnaires and screen capture recordings. Users were required to perform three image search tasks which are specific, general and abstract image search and judge relevance based on ten criteria identified from previous studies. Findings show that some criteria were important when making relevance judgements, with topicality, appeal of information and composition being the common criteria across the search tasks. However the order of importance of the criteria differ between the image search tasks.*

Keywords *Information retrieval, user studies involving documents, Web image search, Relevance criteria, Relevance judgment*

1 Introduction

In the last decade, a large number of digital images have been made available and accessible due to the prevalence of digital imaging technology as well as the growth of the Internet. This has contributed to the development of various image retrieval systems, which in turn has made the process of storing and retrieving images much easier. However, research studies that explore users' relevance judgements for image retrieval are not that common.

Although considerable work has been done in identifying criteria users employ when making text retrieval relevance judgements (for example [1, 7, 12, 14]), little is known about what criteria users employ when making image relevance judgements. Therefore, it is important to explore how users select images in order to develop better retrieval systems with more effective user interfaces.

Relevance is a fundamental notion in information retrieval. Over the years, the field of information retrieval has gained knowledge about relevance, its factors and effects. However, it has mainly focused on traditional textual information retrieval. Relevance, especially in an image is difficult to define satisfactorily. A relevant image is one judged similar in the context of a query. But it depends on the person judging it and in what context is the image relevant. Furthermore, humans are seldom consistent when making judgements. For that matter, there is no way one can guarantee that a user will be consistent in making judgement, especially given the considerable amount of images presented to them. As Volkmer et al. [17] observe, it is difficult to determine whether an image should be judged as relevant or irrelevant, because with an image, there is always room for ambiguity.

The purpose of this exploratory research is to understand people's behaviour when performing image search. The goal is to identify criteria that might be important to a user when they perform image search. Findings from the study will be used to enhance the image search process in order to minimise the users' effort. The rest of the paper is organised as follows. In Section 2 we present some background on users' relevance criteria. In Section 3 we describe the approach and methods used in the study. Results and analysis of the study are discussed in Section 4. Finally we conclude in Section 5 and suggest future work.

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010. Copyright for this article remains with the authors.

2 Related Work

Relevance is an elusive concept that has long been discussed in information retrieval, yet it is still difficult to define clearly. We discuss relevance in Section 2.1 and previous research regarding relevance in the area of image retrieval in Section 2.2.

2.1 Relevance Judgements Criteria

According to Saracevic [13], relevance is not stated, but implied. Different users want different kinds of information. The same information means different things to different people. The same user wants different kinds of information at different times. The same information can mean different things to the same people viewing it at different times. Nonetheless, according to Borlund [3], it can be agreed that relevance involves users' perception of information, at a certain point in time, based on their need situation.

Since the 1990s, there has been a surge of studies on relevance judgement made by real users when given real text retrieval tasks. These studies have been conducted to elicit user's relevance judgement criteria [1, 5, 7, 9, 12, 14, 15]. Saracevic [13] identified these studies as "clues to research". The clues represent artifacts of the search process and the criteria used by the subjects are the attributes which describe these clues. These studies investigated a wide range of criteria and came up with different lists and classifications. For example:

- *accuracy, depth and scope, clarity, recency* [1];
- *authority, accessibility, interesting, topicality, quality* [7];
- *presentation quality, currency, reliability, accuracy* [14].

Although each of the studies were widely varied, they made similar observations about the relevance criteria, which can be generalised as follows [13]:

- Searchers use the same criteria but assign different weights to these criteria.
- The importance of these criteria changes with task, progress in task over time, and varies by some categorisation or class of user.
- Criteria may interact with each other.

However, due to differences between text and image information, users' criteria for image relevance may be very different from textual document relevance judgements.

2.2 Studies on Image Relevance Judgements

Research studies that explore users' relevance judgement on image retrieval are not that common. These studies have explored user's relevance by applying specific information needs and then identifying relevance criteria utilised by the users while making relevance inference [5, 8, 9, 15]. The focus is on criteria users apply while thinking of what is or is not relevant and to what degree it may be relevant.

Choi and Rasmussen [5] conducted a study to observe users' relevance criteria and how these criteria change as expressed before and after the search. Thirty eight faculty and graduate students from the Department of History at Carnegie Mellon University, Duquesne University and the University of Pittsburgh that participated were interviewed. They were using the Library of Congress American Memory photo archives. The authors used nine common criteria which include *topicality, accuracy, time frame, suggestiveness, novelty, completeness, accessibility, appeal of information* and *technical attributes of images.* These criteria were selected from those mentioned by end-users in previous studies. However, they noted that these were not the only criteria and expected users to mention other criteria as well. Users were interviewed to elicit their information need, and they were also asked to rate the importance of each relevance criteria. Information needs were then used by the researchers to perform searches and retrieve images. After providing the participants with the set of retrieved images for their information need, they were once again asked to rate the importance of each relevance criteria. From the results, they observed a significant change in the importance of some criteria across the information seeking process.

Hung et al. [9] investigated the relevance criteria elicited by ten undergraduate students from Department of Journalism and Media Studies at Rutgers University. The participants' relevance judgements were observed by assigning them three different image searches (specific, general, abstract) using the ACCUNET/AP Photo Archive database system. During the search process, participants were asked to save selected photos for later evaluation. After completing all the three search tasks, participants were then interviewed once and asked to describe the relevance criteria that they had used in selecting the photos. Their study identified several common relevance criteria which were used across all three search tasks with *typicality, emotion* and *aesthetic* as the most frequently mentioned.

In a similar follow-up study involving thirty subjects who have photo-editing experience recruited from newspaper and magazine companies, the searchers applied 32 relevance criteria in the specific search, 26 relevance criteria in the general search, and 23 relevance criteria in the subjective search. After comparing the relevance criteria mentioned in the three searchers, 37 types of relevance criteria were identified [8]. This includes the previously identified common criteria [9]. The top ten core criteria were *symbol, composition, consequence, emotion, interest, text, topicality, context, implication* and *facial expression.* His findings also showed that there was a difference in using relevance criteria among the three search tasks.

Sedghi et al. [15] investigated relevance criteria used by twenty six health care professionals when searching for medical images. The participants were asked to specify and perform medical image searches as they would normally do in their daily activities. During the search, they would describe the relevance criteria that they had applied. They found that *visual relevancy, background information* and *image quality* were the three most frequently used relevance criteria. From the interview, they also found that the health care professionals perform image search for different reasons based on their medical image information need. The medical image information need was deemed as the most influential factor in making relevance judgements.

In conclusion, regardless of the experimental setup, users in all these studies apply similar criteria such as *topicality, accuracy/visual relevancy, textual information* and *technical attibutes of images*.

3 Methodology

3.1 Relevance Criteria

During any image search process, it is the user who ultimately decides if the retrieved images are useful or relevant in satisfying their information needs. This decision or assessment of relevance is often influenced by many different criteria. Research by Barry and Schamber [2] suggest that there exist a finite set of criteria which are applied consistently across different types of information users. Although they maybe different in terms of terminology, the criteria seemed to have a common, consistent meaning to users and can also be categorised.

In the relevance criteria study we conducted, we used a subset of the criteria identified in previous image retrieval studies [5, 8]. We selected ten criteria as follows. First, seven criteria (*topicality, accuracy, suggestiveness, completeness, appeal of information, technical attributes of images* and *textual description*) was selected from [5]. We only selected these criteria because they are applicable for all search tasks and not just historical tasks (*time frame* and *novelty*). Secondly, from [8] we selected six criteria (*topicality, composition, consequence, emotion, interest* and *text*). These criteria was selected as they were the core criteria elicited from users when making image relevance judgements for different types of search tasks. Other criteria were not chosen as we did not want to confuse the participants as some criteria can be similar (*symbol, context* and *implication*) or too specific (*facial expression*). Of the thirteen criteria selected from the two studies, three criteria were overlapping. Therefore, for our study, we applied these ten relevance criteria and adapted them for the post-session questionnaires as follows:

1. I selected an image if it was relevant to my search topic (*Topicality*) [5, 8].

2. I selected an image if it was an accurate representation of what I was looking for (*Accuracy*) [5].

3. I selected an image if it gave me new ideas or new insights (*Suggestiveness*) [5].

4. I selected an image if it was interesting (*Appeal of information/interest)* [5, 8].

5. I selected an image if it contained the kinds of details I could use to clarify important aspects of my search topic (*Completeness*) [5].

6. Technical attributes (such as colour, perspective, or angle) were important to me in making my selections for this search task (*Technical attributes of images*) [5].

7. I selected an image if it evoked an emotional response in me regarding the search topic (*Emotion*) [8].

8. Text descriptions of the images were useful in making my selections for this search topic (*Textual information*) [5, 8].

9. I selected an image if it contained consequences or implications of the search topic (*Consequence*) [8].

10. I selected an image if it has strong visual impact (*Composition*) [8].

3.2 Experimental Design

We are interested in understanding users' behaviour when performing image search and aim to identify factors that might be important to a user when they perform image search. Therefore, in designing the experiment, three types of image search tasks were created based on Shatford's image analysis [16]. These include specific, general and abstract image search tasks.

- ***Specific Task***: You are interested in entering a World Cup 2010 contest. One of the contest conditions is that you have to find 6-8 images that best depicts the 2006 World Cup final match in Germany. Your task is to make a selection from a large collection of images from the World Wide Web and save those that in your opinion would most effectively fulfil the contest's condition.

- ***General Task***: As a fashion design student, you are required to create a portfolio showcasing the traditional fabrics of different cultural heritages. Your portfolio will include several different traditional fabrics and one of them is entitled "Timeless Songket". Your task is to make a selection from a large collection of images from the World Wide Web and save 6-8 images that in your opinion would most effectively highlight its uniqueness.

- ***Abstract Task***: You and your classmates are preparing a report on the topic 'Justice and Equality' and your task is to make a selection from a large collection of images from the World Wide Web and save 6-8 images those that in your opinion would most effectively illustrate the meaning of 'justice'.

In our exploratory experiment, we made use of a within-subjects experimental design [10]. We recruited 12 people as volunteers to participate in our study as the subjects of the experiments. All of them are either undergraduate or postgraduate students from RMIT who were approached and recruited via posters, electronic forums and face-to-face recruitment after lecture sessions. The participants were met one at a time, each on a separate occasion. The experiment was conducted anonymously, so that responses could not be traced back to individual participants. For each subject, our procedure was as follows:

1. an introductory orientation session;
2. a pre-search questionnaire;
3. a training session to familiarise the subject on how the task was to be performed;
4. a written instruction for the first task;
5. a search session in which the subject perform the first task;
6. a post-session questionnaire about the first task;
7. steps 4 to 6 were repeated for the remaining two tasks;
8. a final exit questionnaire.

Similar to Hung et al. [9], we used a simulated real work task situation [4] to place our participants in a work task scenario. This scenario allows the participants to fashion their information needs in the same manner as they would when performing an actual search session. The participants were instructed to make a selection of images from the World Wide Web, that in their opinion would be most appropriate for the particular task type. In the course of the search, the participants were allowed to submit as many separate queries as they needed. They could also delete any of the images that they had selected if they changed their mind about the suitability of a particular image. In determining the order of tasks which the participants were to perform, we employed a mathematical factorial design with two users for each of the six permutations of the three tasks. This controls for order effects from learning that participants might acquire from one search task to the next.

The experiment used Google Images[1] search engine to perform image search and retrieval. The experiment was carried out over several weeks and during that time, Google Images changed the way they present image search results. These changes include removing the metadata below the image and having it pop up whenever the user put the cursor on it, which creates a mosaic of images and an infinite scrolling page that presents up to 1000 results per "page" [6]. Only three participants performed their search using the old search interface, while the remaining nine participants performed the tasks using the new interface.

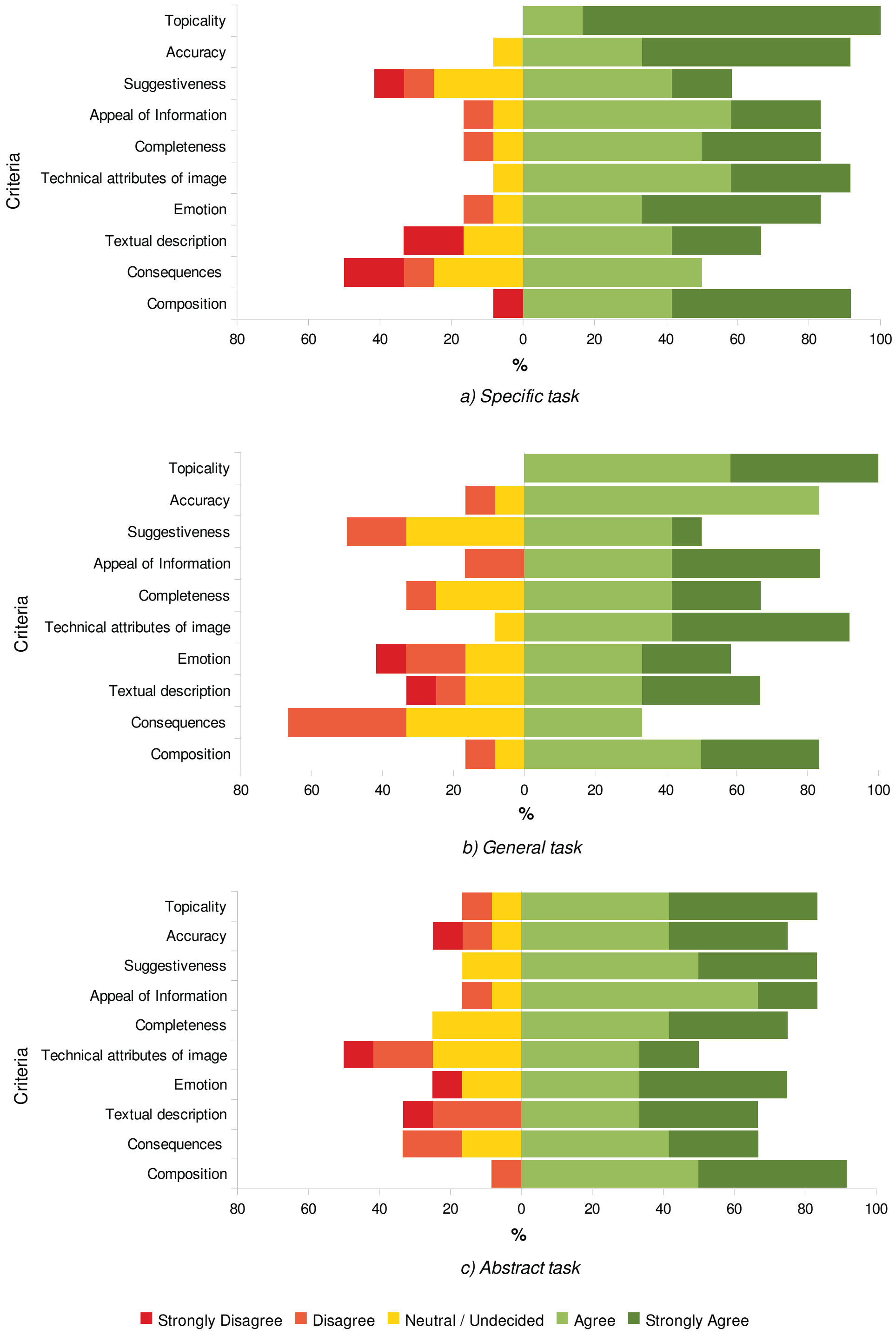
Data for the study was collected through questionnaires and participants' screen capture recordings. Questionnaires were used as it was found to be more effective for users to communicate their response as compared to interview [11]. According to Kelly et al. [11], although users may express more ideas, many of these ideas are similar; they seem to be repeating it rather than providing new ideas. The pre-search questionnaire was used to collect participant's prior experience with image search such as frequently used search engines, search frequency, and search expertise. There were two types of relevance criteria questionnaires: the post-session and the exit questionnaire.

The post-session questionnaire has two sets of closed-ended questions. The first set, asks participants to rate their agreement on the reasons they selected images for the search task that they had just performed based on a selected set of relevance criteria while the second set asked to rate other aspects of the task such as topic familiarity, ease of navigation and result satisfaction. The post-session questionnaire allowed us to collect data and have a better understanding of users' perception of relevance criteria for each task they performed. Finally, open-ended questions were used in the exit questionnaire to collect information regarding the users' whole search experience and any other issues that may have an effect on how they judge image relevance such as what justifies a relevant image, what makes judging relevance difficult (if any) and how to make it easier.

4 Results

Quantitative data from the post-session questionnaires were analyzed using descriptive statistics by assigning numerical values for each agreement rating. This is to determine the average scores of each criteria for relevance judgements and to measure how widely spread the scores were. Another way of showing this information is by calculating the percentage of agreement between users on the cri-

[1] http://www.google.com.au/imghp?hl=en&tab=wi



a) Specific task

b) General task

c) Abstract task

Figure 1: Percentage of agreement between users on the criteria utilised while making relevance judgements for three different type of task.

Table 1: The mean, standard deviation, number of users' in agreement and Chi-Square's p-value for each relevance criteria across search tasks

Relevance criteria	Statistics	Specific Task	General Task	Abstract Task
Topicality	μ	4.83	4.42	4.17
	σ	0.39	0.51	0.94
	# agree	12	12	10
	p-value	**0.0005**	**0.0005**	**0.0209**
Accuracy	μ	4.5	3.75	3.83
	σ	0.67	0.62	1.27
	# agree	11	10	9
	p-value	**0.0039**	**0.0209**	0.0832
Suggestiveness	μ	3.5	3.42	4.17
	σ	1.17	0.9	0.72
	# agree	7	6	10
	p-value	0.5637	1.0000	**0.0209**
Appeal of information	μ	4	4.08	3.92
	σ	0.85	1.08	0.79
	# agree	10	10	10
	p-value	**0.0209**	**0.0209**	**0.0209**
Completeness	μ	4.08	3.83	4.08
	σ	0.9	0.94	0.79
	# agree	10	8	9
	p-value	**0.0209**	0.2482	0.0832
Technical attributes of image	μ	4.25	4.42	3.33
	σ	0.62	0.67	1.23
	# agree	11	11	6
	p-value	**0.0039**	**0.0039**	1.0000
Emotion	μ	4.25	3.5	4
	σ	0.96	1.31	1.21
	# agree	10	7	9
	p-value	**0.0209**	0.5637	0.0832
Textual information	μ	3.58	3.75	3.58
	σ	1.38	1.29	1.44
	# agree	8	8	8
	p-value	0.2482	0.2482	0.2482
Consequence	μ	3.08	3	3.75
	σ	1.16	0.85	1.06
	# agree	6	4	8
	p-value	1.0000	0.2482	0.2482
Composition	μ	4.08	4.42	4.25
	σ	1.14	0.9	0.87
	# agree	11	10	11
	p-value	**0.0039**	**0.0209**	**0.0039**

teria that they find were important when searching, and making image relevance judgement (Figure 1).

Although topicality and accuracy is important across all search tasks, it is more important in the specific and general search with twelve users (100%) for both tasks agree that their image selection was based on the topic of search while eleven users (91.6%) and ten users (83.3%) respectively agree they selected images that is the accurate match of the search. In performing a specific search, the user usually has detailed information about what he/she is looking for. Thus, selecting images that matches the information as accurately as possible. Composition was also a common criteria that users find important across all search tasks. Meanwhile, suggestiveness (83.3%) and consequence (66.7%) is more important, while technical attributes of images is the least important criteria in an abstract search with only six users (50%) as compared to specific (91.6%) and general search (91.7%). A reason for this could be that an abstract image can be represented in so many ways and not easily described like an object, place or action.

On the other hand, it is interesting to learn that a few users would select or judge an image as relevant even if the image does not appeal to them and would not consider technical attributes of the image as an important criteria. This shows that relevance is subjective and each user have different ways of making relevance judgements. The image search tasks was performed on a text-based web search engine by submitting textual queries. Therefore, the returned results will be images that are described by that text. However, across all three image search tasks, there are a few users who disagree that textual description is an important criteria while making relevance judgements. The reason could be that the textual description does not always represent the image that the user was looking for and consequently proved that there is ambiguity when using text to describe images.

In order to examine whether there are statiscally significance differences in the attitudes of the participants in regards to the importance of certain criteria while making image relevance judgements, a Chi-Square analysis was done. The p-value is calculated based on two categories which are (i) combination of strongly agree and agree, and (ii) combination of strongly disagree, disagree and neutral/undecided. For the purpose of this study, it was decided to adopt a minimum significance level of $p<0.05$. Table 1 shows the mean value of each relevance criteria for the three search tasks.

From the table, we can see that the importance of relevance criteria varies between type of tasks and those with higher mean values and number of users who are in agreement (agree and strongly agree) are more widely seen as important when making relevance judgements. This was also shown in the results of the Chi-Square analysis for criteria with a p-value<0.05. It was found that *topicality, appeal of information* and *composition* are important criteria in determining relevance for all search tasks. In contrast, *textual information* and *consequence* are not considered important to users in determining relevance. *Accuracy* and *technical attributes of image* are important for both specific and general tasks. As for the remaining criteria, *suggestiveness* is more important for an abstract search while *completeness* and *emotion* are for a specific task.

As for the exit questionnaire, users were asked to comment on issues regarding image relevance. When asked, “What factors influenced your decision on whether an image was relevant or not”, their responses included: “images related to the topic”; “connection or relationship between image and topic”; “images that reflects the search” and “accurate representation of what I believe the image should look like”. These responses were in accord with responses to another question: “In your opinion, what justifies an image as relevant?”. The users commented: “relevant images should be which will give exact idea about subject of search even if someone doesnt know about it”; “if it describes the topic theme” and “if it is related with the query and it represents the meaning of that query”. Thus, images which satisfy these justifications were considered much more useful or of value and relevant to the users. In addition, users were also asked “Did you find it difficult to decide whether some particular images were relevant or not? If so, what made it difficult?”. All participants agreed that at some point, it can be difficult to decide whether an image is relevant or not and some of their reasons were “sometimes if the query is not true”, “the returned results were not what I expected from the query entered” and “because I knew little or nothing on the topic besides the keywords to search with”. Therefore, although users’ judge relevance based on certain set of criteria, there are other factors that could make passing judgement difficult such as knowledge on the search topic or the context in which the search should be performed. Further analysis on users’ screen capture recordings might reveal more information on how users judge relevance.

Overall, from the ten selected criteria identified from previous studies [5, 8, 9], not all were important to users when judging image relevance. Our results show that users apply more criteria when judging image relevance for specific task as compared to general and abstract task.

5 Conclusion

In this study, 12 participants were aksed to rate their agreement about the relevance criteria that they think is important for searching and selecting specific, general and abstract images. Ten relevance criteria were selected from the criteria set identified from previous studies. The results indicate that users do not find all of the criteria important when making image relevance judgements. Different sets of criteria were used to make relevance judgements for specific, general and abstract images. The three common criteria used were *topicality, appeal of information* and *composition*. However, the order of importance for the criteria differ between the type of tasks. This shows that different search tasks affects how users' judge image relevance. Nonetheless, it is acknowledged that since only one task of each type is used, we may be observing individual task effects rather task type effects. Therefore, further research on a bigger sample with multiple tasks of each type is needed to show the effects of relevance criteria on task type and also to perform statistical tests such as factor analysis for significance of results. Further analysis of results and screen capture recordings will also be done, particularly on the process of users searching and selecting relevant images to find out factors that might have an effect when performing image search.

Acknowledgements This research was made possible with a scholarship and study leave granted by the Universiti Tun Hussein Onn Malaysia and has been approved by the RMIT University Ethics Committee (BSEHAPP 08-10 HAMID). We would also like to thank the students who participated in the study and the anonymous reviewers of this paper.

References

[1] C. Barry. User-defined relevance criteria: An exploratory study. *The American Society For Information Science*, Volume 45, Number 3, pages 149–159, 1994.

[2] C. L. Barry and L. Schamber. Users criteria for relevance evaluation: A cross-situational comparison. *Information Processing and Management*, Volume 34, Number 2–3, pages 219–236, 1998.

[3] P. Borlund. The concept of relevance in IR. *The American Society For Information Science and Technology*, Volume 54, Number 10, pages 913–925, 2003.

[4] P. Borlund and P. Ingwersen. The development of a method for the evaluation of interactive information retrieval systems. *Documentation*, Volume 53, Number 3, pages 225–250, 1997.

[5] Y. Choi and E. M. Rasmussen. Users' relevance criteria in image retrieval in american history. *Information Processing and Management*, Volume 38, Number 5, pages 695–726, 2002.

[6] M. Hachman. Google images gets revamped interface, more relevant results. http://www.pcmag.com/article2/0,2817,2366736,00.asp, July 2010.

[7] S. G. Hirsh. Childrens relevance criteria and information seeking on electronic resources. *The American Society For Information Science*, Volume 50, Number 14, pages 1265–1283, 1999.

[8] T.-Y. Hung. *Search Strategies For Image Retrieval in The Field of Journalism.* Ph.D. thesis, School of Communication, Information and Library Studies, Rutgers University, 2006.

[9] T.-Y. Hung, C. Zoeller and S. Lyon. Relevance judgments for image retrieval in the field of journalism: A pilot study. In *Digital Libraries: Implementing Strategies and Sharing Experiences*, Volume 3815 of *Lecture Notes in Computer Science*, pages 72–80. Springer Berlin/Heidelberg, 2005.

[10] D. Kelly. Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends in Information Retrieval*, Volume 3, Number 1–2, pages 1–224, 2009.

[11] D. Kelly, D. J. Harper and B. Landau. Questionnaire mode effects in interactive information retrieval experiments. *Information Processing and Management*, Volume 44, Number 1, pages 122–141, 2008.

[12] T. K. Park. The nature of relevance in information retrieval: An empirical study. *Library Quarterly*, Volume 63, Number 3, pages 318–351, 1993.

[13] T. Saracevic. Relevance: A review of the literature and a framework for thinking on the notion in information science. Part III: Behavior and effects of relevance. *The American Society For Information Science and Technology*, Volume 58, Number 13, pages 2126–2144, 2007.

[14] L. Schamber. Users criteria for evaluation in a multimedia environment. In *Proceedings of the 54th Annual Meeting of the American Society for Information Science*, pages 126–133, Washington, DC, October 1991.

[15] S. Sedghi, M. Sanderson and P. Clough. A study on the relevance criteria for medical images. *Pattern Recognition Letters*, Volume 29, Number 15, pages 2046–2057, 2008.

[16] S. Shatford. Analyzing the subject of a picture: A theoretical approach. *Cataloging and Classification Quartely*, Volume 6, Number 3, pages 39–62, 1986.

[17] T. Volkmer, J. A. Thom and S. M. M. Tahaghoghi. Exploring human judgement of digital imagery. In *ACSC '07: Proceedings of the thirtieth Australasian conference on Computer science*, pages 151–160, Darlinghurst, Australia, January-February 2007.

A Rule-based Approach for Automatic Identification of Publication Types of Medical Papers

Abeed Sarker
Centre for Language Technology
Department of Computing
Macquarie University
NSW 2109 Australia
abeed.sarker@mq.edu.au

Diego Molla-Aliod
Centre for Language Technology
Department of Computing
Macquarie University
NSW 2109 Australia
diego.molla-aliod@mq.edu.au

Abstract *The medical domain has an abundance of textual resources of varying quality. The quality of medical articles depends largely on their publication types. However, identifying high-quality medical articles from search results is till date a manual and time-consuming process. We present a simple, rule-based, post-retrieval approach to automatically identify medical articles belonging to three high-quality publication types. Our approach simply uses title and abstract information of the articles to perform this. Our experiments show that such a rule-based approach has close to 100% precision and recall for the three publication types.*

Keywords Medical Document Classification, Post-retrieval Classification, Rule-based Classification, Evidence-based Medicine

1 Introduction

Medical practitioners seek high quality information when searching for evidence-based answers to clinical inquiries. The quality of a medical article depends on a number of factors including its publication type. Searching for and appraising high quality articles can be a cumbersome process and requires significant proportions of a practitioner's time when making clinical decisions [5, 7]. This problem is amplified by the large and growing number of available medical articles. The aim of our research is to reduce the time required for the appraisal process by automatic identification of the publication types of medical papers. We propose a simple rule-based approach that uses text from the article titles and abstracts to perform this classification. We show that our proposed approach is extremely efficient at correctly classifying three medical article types (Systematic Reviews, Meta-analyses and Randomized Controlled Trials), which are considered to be of high quality by the medical community, from a set of medical articles belonging to a range of publication types of varying quality levels.

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010. Copyright for this article remains with the authors.

2 Background

2.1 Evidence-based Medicine

Evidence-based medicine (EBM) is the '*conscientious, explicit, and judicious use of current best evidence in making decisions about the care of individual patients*' [14]. Current clinical guidelines urge physicians to practice EBM when providing care for their patients. Good practice of EBM involves finding and appraising current medical evidence before making a decision. Therefore, it involves efficient use of information search and extraction strategies to identify good quality evidence [13].

EBM practitioners require comprehensive, specific bottom-line recommendations that directly answer clinical questions and hence, they often rely on sources of synthesized or pre-appraised evidence. However, databases with synthesized evidence (e.g. Cochrane Library[1]) only cover limited topics and in most cases practitioners have to rely on raw databases such as MEDLINE for information retrieval. MEDLINE, maintained by the US National Library of Medicine (NLM), comprises more than 18 million records and is available online (via the NLM PubMed[2] interface). A typical clinical query on this database returns thousands of results and in most cases assessing the quality of all the returned articles is not possible, particularly at point of care. This is the primary motivation behind implementing a system that performs post-retrieval classification to identify the quality of evidence of medical articles.

2.2 Strength of Evidence and Publication Types

The quality, strength or grade of a recommendation for a clinical query is based on a body of evidence typically found in more than one study. This usually takes into account (i) the level of evidence of the individual studies; (ii) the type of outcomes measured by these studies; (iii) the number, consistency and coherence of the evidence as a whole; and (iv) the relationship between benefits harms and costs [4].

[1] http://www.cochrane.org

[2] http://www.ncbi.nlm.nih.gov/PubMed

The level of evidence of an individual publication is tightly related to the type of publication. Medical publication types include (but are not limited to) Randomized Controlled Trials (RCTs), Systematic Reviews (SRs), Meta-Analyses (MAs), Practice Guidelines, Uncontrolled Clinical Trials, Single Case Studies, Cohort Studies, Tutorial Reviews and even personal opinions. Although all of them provide evidence of some form, the quality of their evidence varies significantly due to the different ways in which the studies are carried out. For example, a clinical trial consisting of a large number of randomly allocated subjects and carried out in a systematic and controlled manner (i.e. a RCT) has a higher level of evidence than a case study of a single patient. In other words, the outcomes presented in the former study are more reliable than the ones presented in the latter.

The connection between the type of publication and the strength of recommendation is generally acknowledged in the numerous grading scales. There are over 100 grading scales in use today [17]. The Strength of Recommendation Taxonomy (SORT) is one such grading scale and it is very popular in EBM practice [4]. SORT provides a uniform recommendation-rating system that can be applied throughout the medicine literature and its simplicity, straightforwardness and comprehensiveness increases its usefulness to practitioners. This taxonomy uses only three ratings A (strong), B (moderate) and C (weak) to specify the strength of recommendation (SOR) of a body of evidence. SORT provides an explicit link between the strength of recommendation and the publication type. Thus, an evidence of grade A may consist of high quality SRs, MAs, RCTs or even cohort studies with good follow-up. Figure 1 shows a pyramid of publication types arranged according to their usual levels of evidence. The pyramid does not explicitly show the SORT grades for the publication types shown. However, for a set of articles, the SORT grade can be derived from the pyramid. Usually, evidence obtained from articles belonging mostly to the top two levels in the pyramid are considered to be of grade A; those from mostly the middle of the pyramid are considered to be of grade B; and those from lower down the pyramid are considered to be of grade C.

2.3 Related Work

During our review of literature in this area, we did not find any previous work attempting to automatically classify medical documents with respect to publication types. However, there has been some research in the area of retrieval of clinically relevant articles. Hunt and McKibbon [9] present some key phrases that are useful for retrieving SRs while Montori et al. [11] use a set of terms including single words or phrases in abstracts or titles, subject headings, publication types etc. The slightly earlier approach proposed by Haynes et al. [8] is similar and relies quite heavily on the metadata associated with each article in MEDLINE instead of the abstract and title texts only. Shonjania and Bero [15] also use metadata for retrieval and provide some PubMed search filters to identify SRs and show them to be quite effective. PubMed also points to some of the above mentioned sources to help practitioners formulate their search queries.

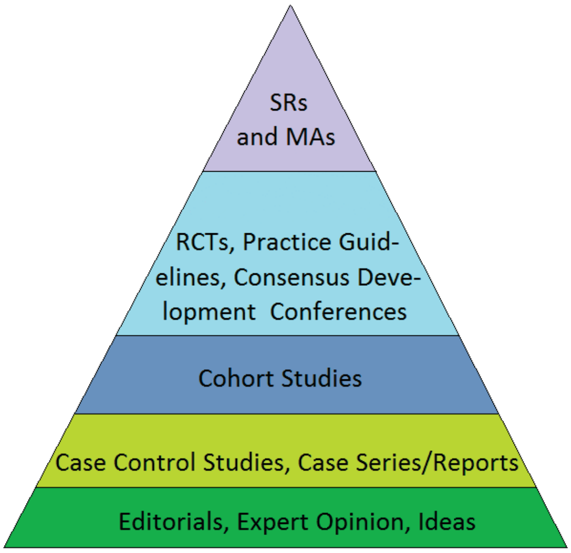


Figure 1: Level of evidence with respect to publication type.

There has also been some research on automatic quality assessment of medical publications. Approaches based on word co-occurrences [6] and bibliometrics [12] have been proposed but these approaches do not integrate EBM recommendations for appraisal. Tang et al. [16], Aphinyanaphongs et al. [1] and Kilicoglu et al. [10] propose approaches for identifying high-quality medical articles that are more relevant for EBM. The post-retrieval re-ranking approach proposed by Tang et al. [16] is not directly comparable to ours because it does not directly take into account medical publication types and is applied to all articles returned by a search engine query (instead of formal, published papers only). Furthermore, their approach is only tested in a very specific sub-domain (i.e. Depression) within the much broader medical domain, which our approach attempts to work in. The other two approaches mentioned above are based on machine learning techniques and are also shown to be quite effective. However, these approaches also rely largely on the metadata accompanying each MEDLINE article. Metadata is only a moderate predictor of the clinical value of an article [3] and relying heavily on metadata associated with a MEDLINE article makes classification approaches suitable for this database only. Additionally, the semi-automatic approach used for indexing MEDLINE articles has evolved with time due to the increasing frequency of medical article publication. As a result, the metadata content may vary significantly between articles published at different times. Furthermore, MEDLINE does not have a 'PublicationType' tag for SRs (they are usually assigned the 'Review' tag together with non-systematic reviews), many articles do not have any 'PublicationType' tag assigned at all and many have multiple distinct tags for this category. An approach that relies solely on the article contents (i.e. titles and abstracts), such as the one we are proposing in this paper, would clearly overcome these problems. This technique can be applied after retrieval to cluster the articles based on their publication types, allowing the practitioner to easily identify the most suitable ones and extract evidence from them.

3 Methods

Abstracts and often titles of medical articles contain information about the types of studies and therefore provide evidence of their publication types. Our approach relies on regular expressions to identify relevant patterns (evidence) from titles and abstracts. At this point of research, we focus only on identifying SRs, MAs and RCTs since articles belonging to these publication types are most often associated with SOR level A, as explained in Section 2.2.

3.1 Rule Development

We developed the expressions used to classify articles by manually studying the titles and abstracts of articles belonging to each of the above mentioned publication types. We collected our development set from a mixture of sources. For articles which have associated 'PublicationType' tags in MEDLINE (e.g. RCTs and MAs) we retrieved about two hundred of each type. We studied each article individually, identified the evidence of publication type and developed patterns to pick up the evidence. During development of the rules, we used an incremental approach similar to the Ripple Down Rules [2] philosophy – after adding a new regular expression we tested its effect on our development set and added more expressions based on the articles that were not correctly identified. For example, in the case of RCTs we primarily developed expressions to detect evidence of randomization in the abstracts. Once evidence of randomization is found, we also developed expressions (from false positives) to detect evidence(s) of unacceptable randomization[3]. Some of the expressions used to identify RCTs are given below:

```
Evidence of Randomization:
'random.*allocate'
'randomi[sz]ed.*study'
'random.*clinical'
'design:.*random'
```

```
Evidence of no or unacceptable
randomization:
'coin\W*flip'
'non\W*random'
'odd\W*even'
'uncontrolled\W*study'
```

For articles without an associated 'PublicationType' tag in MEDLINE (e.g. SRs), obtaining a large development set was considerably difficult. We therefore used a mixture of secondary sources of evidence such as the Journal of Family Practice[4] (JFP) and the Cochrane Library for obtaining about fifty of each and developed our expressions from that set. Furthermore, we studied search techniques suggested by PubMed[5] for efficient retrieval of SRs and developed expressions based on their suggestions. We also developed expressions based on search keywords and techniques suggested in the literature for obtaining articles of specific publication types [9, 11]. After studying the resulting development set, we observed that a relatively small number of carefully developed expressions is sufficient to achieve our goal and in our current approach we use a total of 25 patterns for SRs and MAs and 48 patterns for RCTs.

3.2 Application of Rules

We apply a decision list to identify the publication types of articles. Each article is initially assigned an empty tag and passed through a sequence of tests, each responsible for checking for patterns indicating a specific publication type. At any stage of the sequence, if sufficient evidence of a particular publication type is found (with no further evidence of negation), the article is tagged and removed. The sequence in which the operations are applied is very important as the number of false positives may increase significantly if the sequence is changed. For example, if SRs and MAs are not removed before searching for RCTs, many of the former are falsely tagged as the latter. This is because abstracts of SRs and MAs usually mention the number and types of studies that are being reviewed/analysed, which usually includes RCTs (along with other types of studies). The following list elaborates the actions performed at each stage of the sequence[6]:

1. Check title for evidence of SR or MA[7]
2. Check title for evidence of Practice Guideline or Consensus Development Conference
3. Check title for evidence of RCT
4. Check abstract text for evidence of SR or MA
5. Check abstract text for evidence of RCT

[3] Details about unacceptable randomization techniques for RCTs and other publication types can be found at http://www.nlm.nih.gov/mesh/pubtypes2004.html

[4] http://www.jfponline.com/

[5] The techniques can be found at http://www.nlm.nih.gov /bsd/PubMed_subsets/sysreviews_strategy.html

[6] Steps 2 and 6 are not discussed in this paper

[7] The two types are grouped together since MAs are actually types of SRs.

6. Check for evidence of other low priority publication types (e.g. Evaluation, Cohort Studies, Multicentre Studies etc.)

While checking the abstract of an article for evidence, each sentence is searched separately. We have attempted other approaches such as searching the whole abstract and using a sliding window. However, we have found sentence-level searching to produce the best results primarily because evidence of publication or study type is usually stated or described in a single sentence of an article abstract. Once a pattern match occurs, the entire abstract is searched again to identify patterns that negate the evidence (such as unacceptable randomization techniques in the case of RCTs) and the article is only tagged if no evidence of negation is found. For the mentioned publication types, such a simplistic negation detection technique proves to be sufficient.

4 Results and Discussion

For reasons mentioned in Section 2, we did not depend on the MEDLINE metadata to annotate our test set. We required a set of test articles that were different from the development set and at the same time completely reliable. To achieve this, we used JFP to build our test data. From the Clinical Inquiries sections of the JFP issues, we identified medical articles that are explicitly mentioned (by the JFP authors) to be RCTs, MAs or SRs and were not present in the development set. Importantly, the chosen articles are not actually written by JFP authors, but are cited by them within JFP articles which provide evidence-based answers to clinical queries. Hence, the chosen articles come from a variety of sources and this enables us to test our approach on a diverse article collection. To obtain the article abstracts and titles, we searched for those medical articles in MEDLINE using PubMed and added them to our test set after manually annotating them based on the JFP classifications. Relying on JFP for the test data also allowed us to include articles from a wide range of medical topics, thus ensuring that our approach is not topic dependent. Also, to further prevent bias, all articles identified were added to the test set regardless of their structure/content and the abstracts of the articles were not reviewed during the annotation process. Such a labourious annotation process was necessary due to the lack of substantial reliable annotated data.

For our test set, we used a total of 294 articles including 111 SRs and MAs, 100 RCTs and 83 articles belonging to a mix of other publication types. Including a set of articles belonging to various other publication types was necessary to ensure that our approach does not only correctly tag SRs, MAs and RCTs but also leaves other types of articles untagged. The recall, precision and F-score values are shown in Table 1. For SRs and MAs, our approach produced perfect precision but failed to identify one SR. On the other hand, our approach tagged a total of 97 articles as RCTs, of which 96 were correctly identified.

Publication Type	Recall	Precision	F-Score
MA and SR	0.990	1.00	0.995
RCT	0.960	0.990	0.975

Table 1: Automatic classification results (sample size = 294).

In our post-test review, we discovered that in the case of RCTs, the falsely tagged article was a Review (non-systematic) which mentioned 'one randomized, placebo-controlled study' and was therefore picked up by our rules. As for the four RCTs that were not identified, none of their abstracts contained any evidence of randomization although for one of the RCTs, there was clear evidence of randomization in the full article text. In the case of SRs and MAs, the unpicked article was a SR in which the abstract did not contain any detail of the study type.

The results clearly indicate that a rule-based approach such as ours is very effective in classifying SRs, MAs and RCTs. The high f-scores can be attributed to the fact that articles belonging to these three publication types are very structured (since there are very specific guidelines that must be followed when writing these articles) and therefore their titles and abstracts almost invariably contain sufficient evidence of the type of publication, which can be automatically identified. Furthermore, since the approach does not take into account the metadata associated with each article, it can be applied to articles across various databases.

5 Conclusion and Future Work

In this paper, we have presented an automatic, rule-based approach for classifying medical articles with strong levels of evidence. The results presented here are a step towards a more ambitious goal of automatically identifying the SORs of sets of medical articles. Our results show that the approach is very promising and may be used for automatic classification of other types of medical articles as well. We did not experiment with any machine learning algorithm due to the little amount of annotated data but considering the good results obtained, machine learning is perhaps not necessary. Our future research will focus on testing this rule-based approach with more manually annotated documents. Also, the system will be extended to cover more publication types, which may be a harder problem to solve considering the lower quality of the structure in articles of lower priority.

Acknowledgements This research is jointly funded by Macquarie University and CSIRO.

References

[1] Y. Aphinyanaphongs, I. Tsamardinos, A. Statnikov, D. Hardin and C. F. Aliferis. Text categorization models for high-quality article retrieval in internal medicine. *Journal of the American Medical Informatics Association : JAMIA*, Volume 12, Number 2, pages 207–216, 2005.

[2] Paul Compton and Bob Jansen. Knowledge in Context: A Strategy for Expert System Maintenance. In *Proceedings of the 2nd Australian Joint Artificial Intelligence Conference*, pages 292–306, 1988.

[3] Dina Demner-fushman, Susan Hauser and George Thoma. The role of title, metadata and abstract in identifying clinically relevant journal articles. In *AMIA Annu Symp Proc*, 2005.

[4] Mark H Ebell, Jay Siwek, Barry D Weiss, Steven H Woolf, Jeffrey Susman, Bernard Ewigman and Marjorie Bowman. Strength of recommendation taxonomy (SORT): a patient-centered approach to grading evidence in the medical literature. *Am Fam Physician*, Volume 69, Number 3, pages 548–556, February 2004.

[5] John Ely, Jerome A Osheroff, Mark H Ebell, M. Lee Chambliss, DC Vinson, James J. Stevermer and Eric A. Pifer. Obstacles to answering doctors' questions about patient care with evidence: Qualitative study. *BMJ*, Volume 324, Number 7339, pages 710, 2002.

[6] Thomas Goetz and Claus-Wilhelm von der Lieth. PubFinder: a tool for improving retrieval rate of relevant PubMed abstracts. *Nucleic Acids Research*, Volume 33, pages W774–W778, 2005.

[7] Trisha Greenhalgh. *How to read a paper: The Basics of Evidence-based Medicine*. Blackwell Publishing, 3 edition, 2006.

[8] R Brian Haynes, Nancy Wilczynski, K Ann McKibbon, Cynthia J Walker and John C Sinclair. Developing Optimal Search Strategies for Detecting Clinically Sound Studies in MEDLINE. *J Am Med Inform Assoc*, Volume 1, Number 6, pages 447–458, 1994.

[9] Dereck L Hunt and K Ann McKibbon. Locating and appraising systematic reviews. *Ann Intern Med.*, Volume 126, Number 7, pages 532–538, 1997.

[10] Halil Kilicoglu, Dina Demner-Fushman, Thomas C. Rindflesch, Nancy L. Wilczynski and Brian R. Haynes. Towards automatic recognition of scientifically rigorous clinical research evidence. *J Am Med Inform Assoc*, Volume 16, Number 1, pages 25–31, January 2009.

[11] V. M. Montori, N. L. Wilczynski, D. Morgan, R. B. Haynes and . Optimal search strategies for retrieving systematic reviews from medline: analytical survey. *BMJ*, Volume 330, Number 7482, January 2005.

[12] Maksim Plikus, Zina Zhang and Cheng-Ming Chuong. PubFocus: semantic MEDLINE/PubMed citations analytics through integration of controlled biomedical dictionaries and ranking algorithm. *BMC Bioinformatics*, Volume 7, Number 1, pages 424–439, 2006.

[13] D. L. Sackett, R. B. Haynes, G. H. Guyatt and Tugwell P. *Clinical epidemiology: A basic science for clinical medicine*. Little Brown & Co. Inc., 2 edition, 1991.

[14] David L Sackett, William M C Rosenberg, J A Muir Gray, R Brian Haynes and W Scott Richardson. Evidence based medicine: what it is and what it isn't. *BMJ*, Volume 312, Number 7023, pages 71–72, 1996.

[15] Kaveh G Shonjania and Lisa A Bero. Taking Advantage of the Explosion of Systematic Reviews: An Efficient MEDLINE Search Strategy. *Effective Clinical Practice*, Volume 4, Number 4, pages 157–162, 2001.

[16] Thanh Tang, David Hawking, Ramesh Sankaranarayana, Kathleen Griffiths and Nick Craswell. Quality-Oriented Search for Depression Portals. In Mohand Boughanem, Catherine Berrut, Josiane Mothe and Chantal Soule-Dupuy (editors), *Advances in Information Retrieval*, Volume 5478 of *Lecture Notes in Computer Science*, Chapter 60, pages 637–644. Springer Berlin / Heidelberg, Berlin, Heidelberg, 2009.

[17] Suzanne West, Valerie King, Timothy S. Carey, Kathleen N. Lohr, Nikki McKoy, Sonya F. Sutton and Linda Lux. Systems to rate the strength of scientific evidence. http://www.arhq.gov/clinic/epcsums/strengthsum.htm, April 2002.

Analysis of the effect of negation on information retrieval of medical data

Bevan Koopman[*]
Australian e-Health Research Centre
CSIRO
QLD 4029 Australia
bevan.koopman@csiro.au

Peter Bruza
Faculty of Science & Technology
Queensland University of Technology
QLD 4001 Australia
p.bruza@qut.edu.au

Laurianne Sitbon
Faculty of Science & Technology
Queensland University of Technology
QLD 4001 Australia
laurianne.sitbon@qut.edu.au

Michael Lawley
Australian e-Health Research Centre
CSIRO
QLD 4029 Australia
michael.lawley@csiro.au

Abstract *Most information retrieval (IR) models treat the presence of a term within a document as an indication that the document is somehow "about" that term, they do not take into account when a term might be explicitly negated. Medical data, by its nature, contains a high frequency of negated terms – e.g. "review of systems showed no chest pain or shortness of breath".*

This papers presents a study of the effects of negation on information retrieval. We present a number of experiments to determine whether negation has a significant negative effect on IR performance and whether language models that take negation into account might improve performance. We use a collection of real medical records as our test corpus. Our findings are that negation has some effect on system performance, but this will likely be confined to domains such as medical data where negation is prevalent.

Keywords Information Retrieval, Natural Language Techniques and Documents

1 Introduction

Consider the extract below taken from a patient's medical record:

> "Review of systems is significant for subjective chills and fever with a temperature of 104 this morning. Review of systems is otherwise **negative** for headache, chest pain, shortness of breath, dysuria, or increased frequency of urination." [7, #22248]

Most information retrieval systems would consider queries for "headache" and "chest pain" as good matches for the above document, the assumption being that the presence of a term denotes relevance. For documents that contain little or no negation this may not pose any significant problem, but medical data by its nature contains a high degree of explicit negation [8]. This begs the question of what effect does the prevalence of negation in medical data have on medical information retrieval. Averbuch et al. estimate that ignoring negations in medical narrative reports can reduce retrieval performance by as much as 40% [1].

In this paper we present a number of empirical studies on the effect of negation on current state-of-the-art IR systems. Our test corpus is a collection of medical records and test queries are commonly negated medical terms.

2 Related work

This section summarises some of the work to date on dealing with negation in information retrieval related fields. Much of the focus on negation is in the computational linguistics and NLP fields, less work has focused on negation in retrieval tasks.

This study focuses on explicitly negated terms found in documents and differs from other work concerned with negation in queries, for example the Boolean query "spider AND web NOT internet". Dealing with negation in queries presents its own set of problems, as outlined by McQuire & Eastman [6]. The solution is to exclude documents containing the negated term from the result set. There have been a number techniques to achieve this, these include: post-retrieval filtering [5], negative-scoring, negative-relevance feedback [2] and vector negation [9]. All these approaches focus on negation in the query and do not consider negated terms found in a document.

Prior work for dealing with negation in documents has primarily been done within the Natural Language Processing (NLP) community. The main focus here is on negation detection or recognition – analysing the

[*] Also Faculty of Science & Technology, Queensland University of Technology

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

syntax of natural language to determine which terms have a negative context. A difficult problem is determining the scope of negated terms when negation is detected [4]. This can even prove difficult for human subjects [6]. Many of the solutions to negation detection have been within the application area of dealing with medical data [8, 1], a reflection of the prevalence and importance of negation in medical narratives. NegEx is one popular open source tool for identify negated terms in clinical texts [3]. All these solutions are concerned only with negation detection, they do not propose methods for dealing with negation in the next step of information retrieval.

This paper intends to consider what happens after negation detection. We first provide an empirical analysis of the effect of negation on information retrieval tasks. This is intended to provide the motivation for whether further work on a unified method for negation in IR is justified.

3 Methods

This section provides details of three separate experiments we undertook to investigate the effects of negation on a corpus of medical records.

As our baseline IR system we use the Indri search engine[1] with Porter stemmer for indexing and BM25 term weighting for retrieval. A small comparison of Indri with Lucene showed similar results.

As our test corpus we use the BLULab NLP repository [7], a collection of 81,617 de-identified clinical reports from multiple U.S. hospitals during 2007.

3.1 Experiment A – common negated medical terms

This initial experiment aimed to identify commonly negated terms from the BLULab medical corpus. This was implemented by searching the corpus for the single term appearing after the negation qualifiers: "no", "negative", "negative for" and "not". The number of occurrences matching this pattern for each term was recorded. Terms were then ranked in descending order of the number of negation occurrences.

3.2 Experiment B – precision@10 for negated terms

From the commonly negated terms identified in Experiment A the top 15 (stemmed) terms representing common medical concepts were chosen as candidate queries. These were: *murmur*, *fever*, *fractur*, *edema*, *rash*, *jvd*, *pneumothorax*, *nausea*, *smoke*, *lymphadenopathi*, *mass*, *club*, *wheez*, *headach* and *cyanosi*.

These queries were submitted to the Indri baseline IR system and the top 10 results analysed for their relevance, this gave a measure of precision@10.

3.3 Experiment C – relevance ratio for entire results list

This experiment looked further than precision @ 10 by analysing the entire result set rather than just the top 10 results. The same queries were used as Experiment B (*murmur*, *fever*, etc.). For each query the entire retrieval list was analysed to determine what portion of documents contained the term in negative form and the term in positive form. This gave a relevance ratio for each query q, this is calculated as:

$$\text{rel}(q) = \frac{\text{documents without negation}}{\text{total matching documents}} \quad (1)$$

The experiment was repeated using the top 200 (rather than top 15) negated terms.

A document that contains the term in both positive and negative form would appear in both the lists of positive and negative occurrences for that term.

4 Results

Results of the three experiments are presented in the following subsections. The analysis and interpretation of the results is provided separately in the Discussion, Section 5.

4.1 Experiment A – common negated medical terms

Table 1 presents terms from the BLULab medical corpus that are commonly found in negated form. Terms are ordered in descending frequency of negation occurrences. The terms highlighted in **bold** are the top medical terms chosen as queries for subsequent experiments.

Term	Occurrences	Term	Occurrences
evid	19626	**nausea**	3256
acut	19455	abdomin	3122
have	7951	**smoke**	3115
signific	7856	**lymphadenopathi**	2964
for	7809	had	2883
murmur	6665	short	2793
known	6527	**mass**	2714
other	5722	show	2636
chest	5438	appar	2634
focal	5139	appear	2558
fever	4878	**club**	2510
chang	4690	obvious	2506
fractur	4451	been	2422
histori	4376	activ	2359
edema	4011	**wheez**	2313
be	3953	**headach**	2309
rash	3769	free	2233
jvd	3676	**cyanosi**	2137
definit	3524	abnorm	2035
pneumothorax	3297	prior	2026

Table 1: Commonly negated terms from medical records. Terms in **bold** were chosen as queries.

[1] http://www.lemurproject.org/indri

4.2 Experiment B – precision@10 for negated terms

Table 2 presents precision measures for the top 10 documents returned by each of the 15 queries of commonly negated medical terms. Figure 1 presents these results graphically.

Term	Prec@10	Term	Prec@10
murmur	1.0000	smoke	0.9000
fever	0.9000	lymphadenopathi	0.8000
fractur	0.5000	mass	0.9000
edema	0.9000	club	0.3000
rash	0.8000	wheez	1.0000
jvd	0.3000	headach	1.0000
pneumothorax	0.9000	cyanosi	0.7000
nausea	1.0000		
Average	**0.86**		

Table 2: Precision for top 10 ranked documents for commonly negated medical terms.

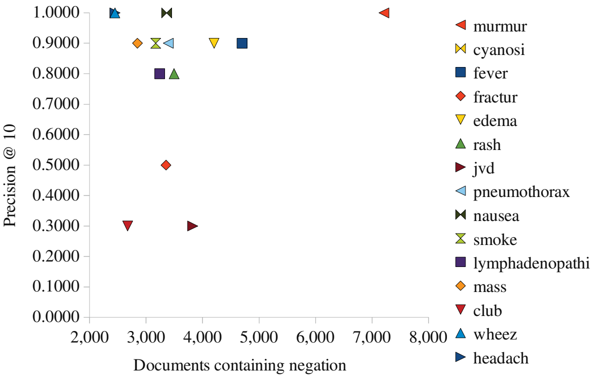


Figure 1: Correlation between negation occurrence and precision @ 10.

4.3 Experiment C – relevance ratio

This experiment presents the relevance ratio – what portion of the entire result set for each query contains the term in positive form. The experiment was done twice, once for the top 15 negated terms and once for the top 200 terms.

Table 3 presents the results for the relevance ratio for the top 15 negated terms. These results are presented in graphical form in Figure 2.

The second part of the experiment was to determine the relevance ratio for the top 200 negated terms, results represented in Figure 3.

Query	Total	Documents with negation	Relevance ratio
murmur	13,573	7,210	0.4688
fever	16,862	4,699	0.7213
fractur	14,194	3,353	0.7638
edema	24,582	4,204	0.8290
rash	7,278	3,495	0.5198
jvd	5,075	3,825	0.2463
pneumothorax	8,428	5,035	0.5974
nausea	15,417	3,365	0.7817
smoke	10,940	3,169	0.7103
lymphadenopathi	7,093	3,241	0.5431
mass	13,569	2,846	0.7903
club	5,823	2,673	0.5410
wheez	6,744	2,448	0.6370
headach	9,322	2,449 3	0.7373
cyanosi	6,649	2,201	0.6690
Average	**11,037**	**3,505**	**0.6371**

Table 3: Relevance ratio – what portion of the entire result set for each query contains the term in non-negated form, see Equation 1, Section 3.3.

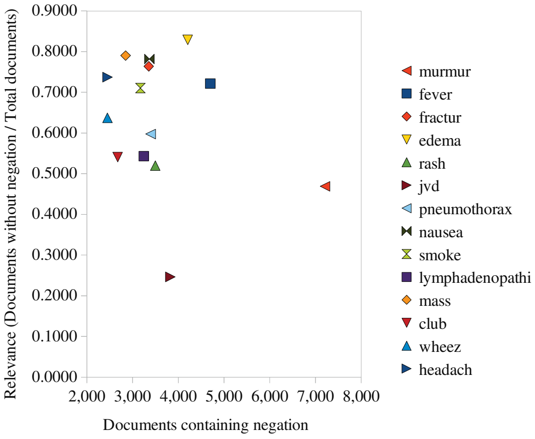


Figure 2: Correlation between negation occurrence and relevance ratio. Top 15 terms.

5 Discussion

The results from Experiment B (see Section 4.2) present the precision @ 10 measurement. Overall the baseline system performs well with an average precision of 0.86. In most cases documents containing the negated form were not found in the top 10 results. The reason for this is that when a term occurs in negated form it typically only occurs once within the document, for example a document will have a single mention of “no rash”. In contrast when the term appears in positive form it typically appears a number of times – a medical record relating to someone suffering from a rash will mention the term “rash” multiple times. The standard term-weighting function will rank the document containing multiple positive occurrences of “rash” above that of the single negative occurrence. In this way current IR systems implicitly deal with negation by their standard document / term frequency weighting functions.

In Experiment C we considered the entire result set returned (rather than just the top 10 documents). Here negation had a more marked affect, average precision was 0.6371. However, there was no strong correlation between the occurrence of negation and performance (as shown in Figure 2). “JVD” and “murmur” were two queries that performed well below the average, these

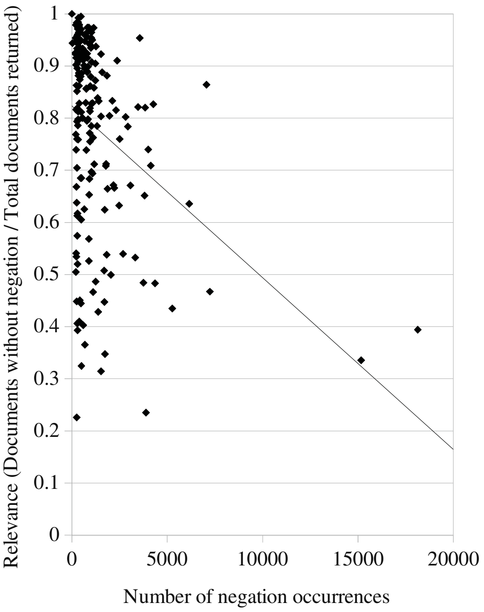


Figure 3: Correlation between negation occurrence and relevance ratio. Top 200 terms.

two terms are part of a standard observation doctors perform on all patients and therefore nearly always appear in a patient's record in negated form.

Overall negation does not have a major impact on retrieval. Term weighting functions are effective at ranking documents with negated terms. We conclude that specific methods of dealing with negation would only be required for specific domains such as medical data where negation is prevalent and may pose problems in the quality of results retrieved.

5.1 Limitations & future work

In our experiments negation detection was implemented by searching the corpus for the single term appearing after the negation qualifiers: "no", "negative", "negation for" and "not". This simplistic approach would not identify more complex examples such as "history inconsistent with stroke" or "patient denies any pain". Additionally we do not identify negated conjunctions like the example presented in the introduction – "...negative for headache, chest pain, shortness of breath, ...". We would only identify "headache" as a negated term from this extract.

Implementing a best-practise NLP negation detection tool (e.g. NegEx) would likely increase the negative effects of negation on the relevance ratio results (Experiment C). It is, however, unlikely to affect the precision @ 10 results, which we believe is the more important indicator.

6 Conclusion

We have presented medical data as a domain where negation in documents is prevalent. Based on this we have conducted a number of experiments to determine what effect the high prevalence of negation has on information retrieval. The purpose of which is to determine whether specific methods of dealing with negation might be developed to improve retrieval performance. Our findings are that modern term-weighting functions used in IR systems are quite effective at dealing with negation and that specific methods for dealing with negation are only really relevant to specific domains such as dealing with medical data.

7 Acknowledgements

This study was supported by the Australian e-Health Research Centre. We thank the University of Pittsburgh BLULab group [7] for providing the test corpus of medical records.

References

[1] Mordechai Averbuch, Tom H. Karson, Benjamin Ben-Ami, Oded Maimond and Lior Rokachd. Context-sensitive medical information retrieval. In *Proceedings of the 11th World Congress on Medical Informatics (MEDINFO-2004)*, San Francisco, USA, 2004.

[2] Mark D. Dunlop. The effect of accessing non-matching documents on relevance feedback. *ACM Transactions on Information Systems*, Volume 15, Number 2, pages 137 – 153, 1997.

[3] Ilya M. Goldin and Wendy W. Chapman. Learning to detect negation with 'not' in medical texts. In *Workshop at the 26th ACM SIGIR Conference*, Toronto, Canada, 2003.

[4] Laurence R. Horn. *A natural history of negation*. University of Chicago Press, Chicago, IL, 1989.

[5] Gerard Saltonand Michael J. McGill. *Introduction to modern information retrieval*. McGraw-Hill Book Company, New York, 1984.

[6] April R. McQuire and Caroline M. Eastman. The ambiguity of negation in natural language queries to information retrieval systems. *Journal of the American Society for Information Science*, Volume 49, Number 8, pages 686 – 692, 1998.

[7] University of Pittsburgh. BLULab NLP Repository. http://nlp.dbmi.pitt.edu/nlprepository.html, July 2010.

[8] Lior Rokach, Roni Romano and Oded Maimon. Negation recognition in medical narrative reports. *Information Retrieval*, Volume 11, pages 499 – 538, 2008.

[9] Dominic Widdows. Orthogonal negation in vector spaces for modelling word-meanings and document retrieval. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, Morristown, USA, 2003.

Rule-based Approach for Identifying Assertions in Clinical Free-Text Data

Yue Kimi Sun
Faculty of Science & Technology
Queensland University of Technology
QLD 4001 Australia
Kimi.Sun@csiro.au

Anthony Nguyen
The Australian e-Health Research Centre
CSIRO
QLD 4029 Australia
Anthony.Nguyen@csiro.au

Laurianne Sitbon
Faculty of Science & Technology
Queensland University of Technology
QLD 4001 Australia
Laurianne.Sitbon@qut.edu.au

Shlomo Geva
Faculty of Science & Technology
Queensland University of Technology
QLD 4001 Australia
S.Geva@qut.edu.au

Abstract *A rule-based approach for classifying previously identified medical concepts in the clinical free text into an assertion category is presented.There are six different categories of assertions for the task: Present, Absent, Possible, Conditional, Hypothetical and Not associated with the patient. The assertion classification algorithms were largely based on extending the popular NegEx and Context algorithms. In addition, a health based clinical terminology called SNOMED CT and other publicly available dictionaries were used to classify assertions, which did not fit the NegEx/Context model. The data for this task includes discharge summaries from Partners HealthCare and from Beth Israel Deaconess Medical Centre, as well as discharge summaries and progress notes from University of Pittsburgh Medical Centre. The set consists of 349 discharge reports, each with pairs of ground truth concept and assertion files for system development, and 477 reports for evaluation. The system's performance on the evaluation data set was 0.83, 0.83 and 0.83 for recall, precision and F1-measure, respectively. Although the rule-based system shows promise, further improvements can be made by incorporating machine learning approaches.*

Keywords rule-based, medical concept, assertion, NegEx, Context, SNOMED CT.

1 Introduction

A large part of clinical data is recorded in natural language, which makes algorithmic processing by a computer a very hard task. Three sequential tasks defined by the i2b2 NLP Challenge [1] consist of Concept Annotation, Assertion Annotation and Relation Annotation, which are three fundamental steps for processing clinical data. The Concept Annotation task builds toward the Assertion and Relation tasks of the challenge. This means that, the output of the Concept task is used as input to the Assertion task, and the output of both the Concept and Assertion task can be used for the Relation task.

In this paper, only the Assertion Annotation task was studied. In the context of the i2b2 NLP Challenge, an Assertion is defined as a contextual attribute (either 1. Present, 2. Absent, 3. Possible, 4. Conditional, 5. Hypothetical or 6. Not associated with the patient) that is applied to a concept relating to a medical problem.

2 System Description

The system was developed using GATE [1], an open source framework for developing and deploying software components that process natural language. Figure 1 shows the architecture of the assertion classification system. It consists of three stages, namely: 1) Preprocessing, 2) Assertion relevance matching, and 3) Assertion generation.

The system was largely based on a popular regular expression based negation/context algorithm [2, 3], which has been proven to work well with clinical free text data. Additional algorithms were also developed to accommodate assertions that cannot be classified using the NegEx/Context approach.

[1] Fourth i2b2/VA Shared-Task and Workshop Challenges in Natural Language Processing for Clinical Data. https://www.i2b2.org/NLP/Relations/

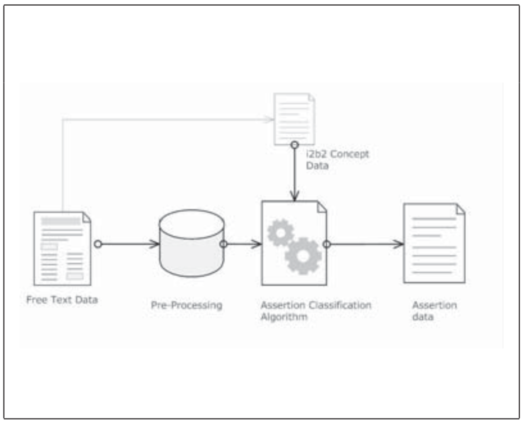
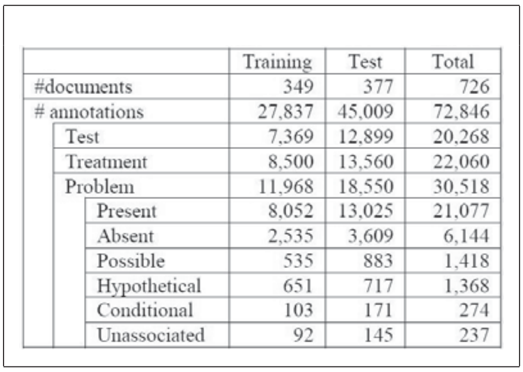


Figure 1: Assertion classification system.

For the Assertion Annotation task, the system is required to generate an assertion category for each concept identified as a medical problem. The input concept data is assumed to be available by the assertion classification system. For the purposes of system development and evaluation, the concept data is provided by human experts for each team. The problem of categorizing concepts into assertion classes is a typical classification task. Figure 2 shows the corpus statistics to the assertion classification task, there were 11968 concepts relating to medical problems in training data for system development, with another 18550 concepts which were used for testing.



	Training	Test	Total
#documents	349	377	726
# annotations	27,837	45,009	72,846
Test	7,369	12,899	20,268
Treatment	8,500	13,560	22,060
Problem	11,968	18,550	30,518
Present	8,052	13,025	21,077
Absent	2,535	3,609	6,144
Possible	535	883	1,418
Hypothetical	651	717	1,368
Conditional	103	171	274
Unassociated	92	145	237

Figure 2: Corpus statistics to assertion classification task

2.1 Preprocessing

The preprocessing step performs the tagging of entities such as tokens, sentences and concepts which were required for the assertion relevance matching stage.

The tokeniser splits the text into simple tokens which were separated by a space. Sentences were separated by line breaks, since this was the general structure in which the reports were formatted. These tokens and sentence annotations were used to annotate the i2b2 input concept data.

Although, the tokeniser and sentence splitter were simplified for this task, in practice more sophisticated algorithms would be required to distinguish sentence boundaries from tokens such as decimal numbers, punctuations and abbreviations. Automatically mapping medical concepts from free text would also be required in practice, since concept annotations are generally not available. A number of medical concept annotators exist, however, their performance may vary [4, 5].

2.2 Contextual analysis

We hypothesized that each assertion category could be largely classified using the methodology adopted in NegEx [2] or more generally the Context [3] algorithm. Context identifies common assertions phrases in the free text, and subsequently applies the respective assertion to a concept (or indexed term) based on a regular expression based template and the type of assertion phrase that was found.

Two types of assertion phrases were defined, namely, pre-assertion and post-assertion phrases. Pre-assertion phrases occur before the term (or concept) they assert, while the post-assertion phrases occur after the term they assert. For example, "pre-assertion" phrases would apply to concepts appearing after the assertion phrase (e.g., the sentence "The patient <pre-negation>denies<pre-negation><concept>chest pain<concept>", would assert the concept "chest pain" as "absent"), and vice versa for "post-assertion" phrases. The scope of search for concepts to apply the assertion was bounded by conjunction phrases and/or sentence boundaries.

The list of assertion phrases used in Context was extended and updated using examples from the i2b2 development data set. This demands a lot of knowledge about the domain language itself to correctly identify assertion phrases.

The algorithm was also extended to incorporate possibility phrases which assert uncertainty between two concepts. An example of a possibility phrase commonly occurring between two concepts is "versus" (or its variants). In such a case, the two concepts appearing before and after the possibility phrase would both be asserted as "possible".

2.3 Self asserted concepts

Although the algorithm above would associate concepts with assertions according to the context surrounding the concept, it cannot classify assertions to concepts when the meaning of morphology of the concept implies the assertion. For example, concepts such as "afebrile" and "nontender" would be considered "self-asserted" concepts and be classified as an absent assertion. To address this limitation, the health based ontology SNOMED CT (Systematized Nomenclature of Medicine Clinical Terms) [6] and publicly available dictionaries were incorporated. SNOMED CT is a systematically organized computer processable collection of medical terminology covering diseases, findings, procedures, pharmaceuticals etc. Among these, the concept "Clinical Finding Absent" was used to test if it subsumes (or is an ancestor of) medical concepts found in the free text. If subsumed, then the concepts would be asserted as absent. An in-house ontology server was used to query the subsumption relationships.

In addition, publicly available dictionaries from Internet were incorporated to further identify self-asserted concepts. A public resource from the Internet [8], which consists of 31 English dictionaries (covering 869,228 words or terms), was included in the system. It was conjectured that concepts containing known prefixes representing an absent assertion such as “non” would contain a stem of a word when the prefix was removed. If the stem of the concept is found in the dictionary, then the concept would be considered a “self-asserted” concept and be classified as an absent assertion.

2.4 Post Processing

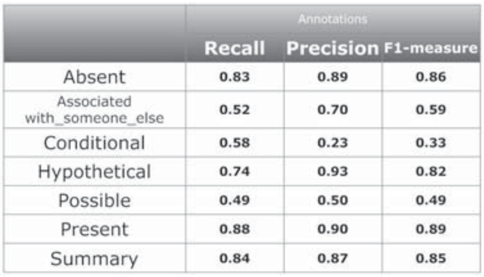
Post-processing of the assertions was performed to ensure that each concept contains only a single class of assertion. If more than one class of assertion exists for a given concept, the choice of assertion was selected depending on a priority list given by:

1. Not associated with the patient
2. Hypothetical
3. Conditional
4. Possible
5. Absent
6. Present

3 System Evaluation

The i2b2 / VA Challenge data set consists of 349 discharge reports, each with pairs of ground truth concept and assertion files for system development, and 477 reports for evaluation.

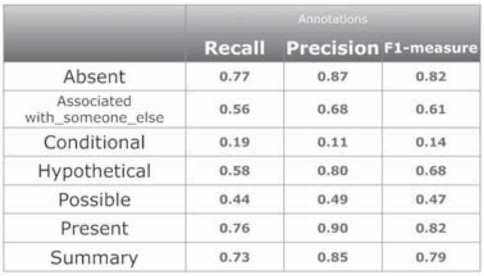
The system was evaluated using recall, precision and F1-score measures.



	Annotations		
	Recall	Precision	F1-measure
Absent	0.83	0.89	0.86
Associated with_someone_else	0.52	0.70	0.59
Conditional	0.58	0.23	0.33
Hypothetical	0.74	0.93	0.82
Possible	0.49	0.50	0.49
Present	0.88	0.90	0.89
Summary	0.84	0.87	0.85

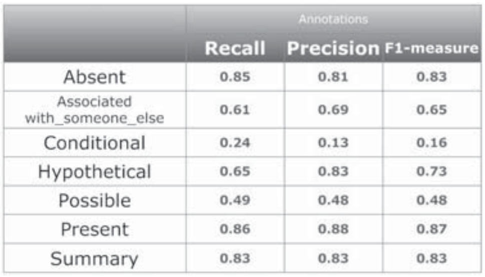
Figure 3: Overall System Performance on training data

Overall performance on the 2010 i2b2 /VA Challenge training corpus of 349 discharge reports against a database of ground truth assertion decisions are shown in Figure 3, and resulted in a recall, precision and F1-measure of 0.84,0.87, and 0.85, respectively.



	Annotations		
	Recall	Precision	F1-measure
Absent	0.77	0.87	0.82
Associated with_someone_else	0.56	0.68	0.61
Conditional	0.19	0.11	0.14
Hypothetical	0.58	0.80	0.68
Possible	0.44	0.49	0.47
Present	0.76	0.90	0.82
Summary	0.73	0.85	0.79

Figure 4: System Performance on testing data by only use Contextual Analysis



	Annotations		
	Recall	Precision	F1-measure
Absent	0.85	0.81	0.83
Associated with_someone_else	0.61	0.69	0.65
Conditional	0.24	0.13	0.16
Hypothetical	0.65	0.83	0.73
Possible	0.49	0.48	0.48
Present	0.86	0.88	0.87
Summary	0.83	0.83	0.83

Figure 5: Overall System Performance on testing data

The performance on the testing data are shown in Figures 4 (Contextual analysis only) and Figure 5 (Contextual analysis and self-assertions). The performance of Contextual analysis algorithm on the 2010 i2b2 /VA Challenge test corpus of 477 discharge reports against a database of ground truth assertion decisions were 0.73, 0.85, and 0.79 for recall, precision and F1-measure, respectively. This is the baseline performance for the core NegEx and Context algorithms, which didn’t include the processing of self asserted concepts as described in section 2.3.

The performance improves further when self-asserted concepts are incorporated. Overall performance on the test corpus were 0.83, 0.83, and 0.83 for recall, precision and F1-measure, respectively. While the performance of the system shows promise, the methodology could be much improved to enhance the performance of the less prevalent assertion classes.

4 Possible Improvements

The proposed rule-based system shows promise but is limited in performance compared with the best performing Supervised or Hybrid systems, which can perform up to 0.93 for recall, precision and F1-measure.

The contextual analysis based algorithm is limited to the list of assertion phrases known to the system and unable to always make linguistic sense or are consistent with various types of semantic constraints. New unseen phrases will therefore be overlooked and result in misclassifications. The assertion phrases themselves are also subject to a trade-off between recall and precision. Significant knowledge about the domain language itself to correctly identify assertion phrases is thus necessary. For example, one word could completely change the sense of a statement. The statement could then be inverted, weakened or amplified. The following simple example by Horn [7] shows this effect in negated sentences:

1. I'm not tired.

2. I'm not a bit tired. (which equals "I'm not at all tired.")

3. I'm not a little tired. (which equals "I'm quite tired.")

The algorithm can also be extended to take into account of the low-level POS (Part of Speech) and grammatical sentence structure and/or use machine learning based approaches such as Conditional Random Fields (CRF) to learn the association between the phrases in the free text and the possible assertions that they represent. One the other hand, active learning methods maybe useful for selectively sampling (as opposed to randomly sampling) from a large corpus for tagging using various entropy-based scores [9].

5 Conclusion

A simple rule-based approach for classifying previously identified medical concepts in the clinical free text into an assertion category was proposed and shows promise. Further improvements can be made by incorporating machine learning approaches to learn the associations between concepts and assertions that are difficult to achieve with rule-based approaches.

Acknowledgements The data for this project includes discharge summaries from Partners HealthCare and from Beth Israel Deaconess Medical Center (MIMIC II Database), as well as discharge summaries and progress notes from University of Pittsburgh Medical Center. We would like to thanks for their contributions to this task.

References

[1] Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Marin Dimitrov et al. Developing Language Processing Components with GATE Version 5.2009;1:17-25.

[2] Chapman WW, Bridewell W, Hanbury P, Cooper GF, Buchanan BG. A simple algorithm for identifying negated findings and diseases in discharge summaries. J Biomed Inform. 2001, 34:301-10.

[3] Henk Harkema, John N. Dowling, Tyler Thornblade, Wendy W. Chapman, ConText: An algorithm for determining negation, experiencer, and temporal status from clinical reports, Journal of Biomedical Informatics, Volume 42, Issue 5, Biomedical Natural Language Processing, October 2009, Pages 839-851.

[4] Aronson, AR, Effective mapping of biomedical text to the UMLS metathesaurus: The MetaMap program, American Medical Informatics Association (AMIA) Annual Symposium, pp. 17-21, 2001.

[5] Nguyen AN, Lawley MJ, Hansen DP, Colquist S., A Simple Pipeline Application for Identifying and Negating SNOMED Clinical Terminology in Free Text. Health Informatics Conference, Canberra, Australia. pp. 188-193, August 2009.

[6] International Health Terminology Standards Development Organisation , SNOMED CT, http://www.ihtsdo.org/snomed-ct/

[7] Laurence R. Horn. A natural history of negation. University of Chicago Press, Chicago, Illinois, June 15 1989.

[8] http://www.linux-pour-lesnuls.com/traduc/Dictionnaires/

[9] Tang, M, Luo, X, and Roukos, S. Active learning for statistical natural language parsing, in Proc. of the 40th Annual Meeting of the Association for Computational Linguistics pp 120-127 July 7-12 Philadelphia, PA 2002.

Confidence Intervals for Information Retrieval Evaluation

Laurence A. F. Park
School of Computing and Mathematics
University of Western Sydney, Australia
lapark@scm.uws.edu.au

Abstract
Information retrieval results are currently limited to the publication in which they exist. Significance tests are used to remove the dependence of the evaluation on the query sample, but the findings cannot be transferred to other systems not involved in the test. Confidence intervals for the population parameters provide query independent results and give insight to how each system is expected to behave when queried. Confidence intervals also allow the reader to compare results across articles because they provide the possible location of a systems population parameter. Unfortunately, we can only construct confidence intervals of population parameters if we have knowledge of the evaluation score distribution for each system. In this article, we investigate the distribution of Average Precision of a set of systems and examine if we can construct confidence intervals for the population mean Average Precision with a given level of confidence. We found that by standardising the scores, the system score distribution and system score sample mean distribution was approximately Normal for all systems, allowing us to construct accurate confidence intervals for the population mean Average Precision.

Keywords Information Retrieval, Evaluation

1 Introduction

When publishing information retrieval system evaluation results, the mean score from a sample set of queries is reported. These results are usually presented with the confidence in hypothesis test results when compared with a baseline system. Reporting the sample mean allows the reader to compare the presented set of systems for the given set of queries, while the hypothesis tests indicate how well the results generalise to a new sample of queries.

Unfortunately, there is no method for comparing systems across publications. We are able to compare the sample mean scores, but by doing so we have no indication of how the systems will perform when given a new sample of queries. Results from hypothesis test report the confidence in the test, and therefore the tests information cannot be used to compare systems across publications. The reader's only option is to obtain the set of systems in each publication and run experiments to identify if one is more accurate than the other.

By having knowledge of a population parameter, such as the population mean evaluation score for a system, we would be able to compare systems independent of the sample set of queries used. We are unable to compute an exact value for the population mean using a sample set of queries, but we are able to construct a confidence interval, giving a range in which the population mean evaluation score is most likely to exist.

To compute accurate confidence intervals for a population parameter from samples, we must have knowledge of the distribution of the associated sample statistic. In this article, we investigate the distribution of Average Precision and the sample mean Average Precision to compute accurate confidence intervals for the population mean Average Precision.

We make the following important contributions:

- An investigation into how we can report the confidence intervals for the population mean Average Precision (Section 4 and 5).
- A description on how results should be reported to allow others to reuse the results (Section 6).

The article will proceed as follows: Section 2 provides a brief overview of Information Retrieval evaluation, Section 3 discusses the portability of published Information Retrieval results, Section 4 examines the distribution of Average Precision results and identifies if we are able to construct accurate confidence intervals. Section 5 examines the effect of standardisation of the distribution of Average Precision results. Finally, Section 6 presents further details of the confidence interval we have found.

2 System evaluation

First let us define the retrieval system. A retrieval system is a function $S(q, D)$ on query q and document set D, where $S : q \times D \rightarrow \mathbb{R}^N$. The output of the function S is a vector $\vec{r}_{q,D} = \{r_{q,d_1}, r_{q,d_2}, \ldots, r_{q,d_N}\}$ containing a weighted list, where each weight r_{q,d_i} is associated to the relevance of document i in D to query q.

An evaluation measure is a function $m_{q,D} = E(\vec{r}_{q,D}, \vec{\rho}_{q,D})$ on the weighted document list $\vec{r}_{q,D}$ and the set of true relevance judgements $\vec{\rho}_{q,D}$, where $E : \mathbb{R}^N \times \mathbb{R}^N \rightarrow \mathbb{R}$. The output of E is a scalar value

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

which reflects the accuracy of system S on document set D using query q.

To truly test the accuracy of a system on a document collection, we would obtain all of the queries that will be used, along with their probability of use, and compute the expected system accuracy using

$$\mathbb{E}[s_D] = \sum_{q \in \Phi} E(\vec{r}_{q,D}, \vec{\rho}_{q,D}) P(q)$$

where $P(q)$ is the probability of issuing query q and Φ is the population of queries.

Two problems exist with this form of evaluation. First, the population of queries Φ depends on the future use of the system. We could obtain an estimate of Φ by releasing the system and recording all of the queries that are used, but there is no way of knowing how good an estimate this is. Second, for each query we need a set of relevance judgements for document set D. For one query, this requires manually judging the relevance of all documents in D. If D contains one million documents, we must perform one million relevance judgements. For k queries, we must perform k million judgements.

To overcome the first problem, the information retrieval community has resorted to using a sample of queries and treating each query as being equally likely. This changes the expectation equation to simply computing the mean of a sample:

$$\overline{m}_D = \sum_{q \in Q} E(\vec{r}_{q,D}, \vec{\rho}_{q,D}) \frac{1}{k}$$

where $Q \subset \Phi$, k is the cardinality of set Q, and $\overline{m}_D$ is the sample mean system score over the query sample set Q. The sample mean is used as an estimate of the population mean (expected value), but estimates of how well this is approximated are not provided in experimental results.

To overcome the second problem, methods such as pooling [3] can be used to reduce the load of this task, but significant effort must still be placed into this process.

By themselves, the sample mean evaluation scores are limited in their use. The sample mean scores are used in most retrieval experiments to compare against the sample mean retrieval scores of another system, where both systems are evaluated using the same sample.

To remove the dependence of the evaluation on the query sample, a paired hypothesis test (e.g. the Wilcoxon signed rank test) for an increase in evaluation score can be performed for a pair of systems. The result from the test is the level of confidence of the first system providing a greater score than the second for a randomly sampled query.

3 Portability of results

We showed in the previous section that we are able to compare two retrieval systems using a paired significance test. To conduct the test, we require the evaluation score for each system for a specific set of queries. Therefore, if we have access to both systems S_x and S_y, and we have a document set D, a random sample of queries Q and the associated relevance judgements, we simply generate the system score using a suitable evaluation metric E and compare the paired evaluation scores using a significance test.

If a reader obtains two publications that have developed new systems S_x and S_y respectively, the reader is unable to determine from the published results in both articles if there is any statistically significant difference in results between systems S_x and S_y. The reader should be able to compare the sample means of each system from each article as an estimate of the expected performance of each system, but the reader would have no knowledge of the accuracy of the estimation. Paired significance tests would be provided in each article, but the paired test results only apply to the systems involved in the test and give no indication of how the system compares with others not involved in the test.

At the moment, the only way to compare two systems that appear in separate publications is to obtain the systems and run our own experiments. This implies that the current method of reporting information retrieval results limits the evaluation to the publication. We are unable to compare retrieval evaluations across articles and therefore our results are not portable.

To provide portable results, all retrieval experiments should provide details of system population parameters. Population parameters provide details on how the evaluated system behaves independent of the query sample used and can also provide us with information such as the expected evaluation score for the system.

Since system population parameters are independent of the query sample, we are able to compare the values of multiple systems across different publications, making the results portable.

If we obtain a sample from a given population, we are not able to compute the exact value of population parameters, but we can compute a confidence interval for a population parameter using statistical methods. Therefore, if we have a set of evaluation scores for a given system obtained from a sample set of queries, we are able to compute a confidence interval for a certain population parameter.

For each confidence interval, we need an associated confidence level, where the confidence level is related to the probability of a Type I error occurring (the probability of the population parameter not being in the interval). For a confidence interval to be useful, the probability of a Type I error should be low.

To accurately compute the probability of a Type I error for a given confidence interval, we need to know the distribution function associated to the sample data. Therefore to compute the confidence level of a confidence interval for a given system, we need to know the distribution function of the evaluation score distribution. To the best of our knowledge, there has been no study into the distribution of retrieval system evaluation scores.

In the following sections we will investigate the distribution of Average Precision over a set of systems and identify how we can use the distribution to construct a confidence interval for the population mean Average Precision with an associated accurate measure of Type I error.

A system's population mean evaluation score is the expected score for a randomly sampled query. This parameter is of interest because it provides us with a measure of how well the system will perform when provided with an unknown query. There has been much research into computing the confidence interval of the population mean for given distributions, therefore we will use the knowledge from the prior research and identify how well it applies to a set of system distributions.

To compute the confidence interval for a system population mean Average Precision (AP), we must

1. identify the distribution of the sample mean AP,
2. compute an estimate of the parameters of the sample mean AP distribution given the sample,
3. finally, identify the quantiles of the distribution that contain the desired level of confidence.

4 Average Precision Distribution

To test the validity of confidence interval experiments, we require knowledge of the population statistics of the system score distributions. A system score distribution is the probability of obtaining a particular score from a randomly sampled query for the given retrieval system on a given document set. System score distributions have not been computed or approximated for any retrieval system (to the best of our knowledge). Therefore we will approximate a set of system score distributions using the scores from a large sample of queries.

In this article, we have used the system scores from the TREC 2004 Robust track. The TREC Robust track contains 249 queries and results from 110 retrieval systems on a document collection containing $528,155$ documents from TREC disks 4 and 5 (excluding the Congressional Record document set).

We will use the following notation:

- AP is the Average Precision from a sample query for a given system,
- $\overline{\text{AP}}$ is the sample mean Average Precision for a given system from a sample of n queries (usually known as mean Average Precision),
- μ_{AP} is the population mean Average Precision for a given system,
- s_{AP} is the sample standard deviation Average Precision for given system from a sample of n queries,
- σ_{AP} is the population standard deviation of Average Precision for a given system,

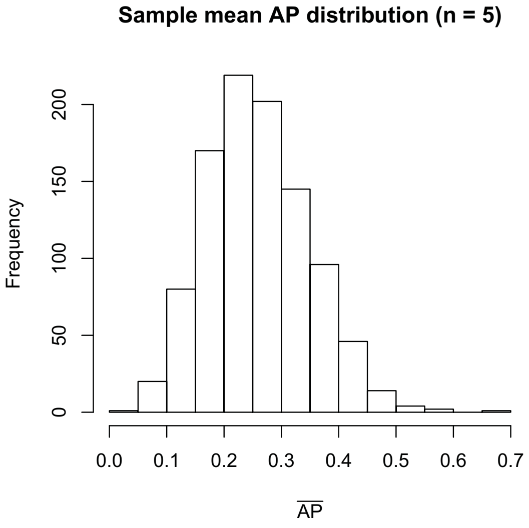


Figure 1: Distribution of a randomly sampled system's sample mean ($\overline{\text{AP}}$) using $n = 5$.

- $\sigma_{\overline{\text{AP}}}$ is the population standard deviation of the sample mean Average Precision for a given system.

Using the TREC Robust data, we are able to estimate the population parameters using the set of 249 queries and the sample statistics using a smaller subset of the queries. For example, μ_{AP} is computed for a given system by computing the mean across all 249 queries, while $\overline{\text{AP}}$ is computed using a small subset of the queries (such as $n = 10$).

4.1 Confidence when σ_{AP} is known

In this section we will examine the accuracy of a confidence interval under the assumption that AP follows a Normal distribution and σ_{AP} is known for each system.

The Central Limit Theorem [2] tells us that given a Normally distributed random variable x with mean μ and standard deviation σ, its sample mean $\overline{x}$ is also Normally distributed with mean μ and standard deviation $\sigma/\sqrt{n}$, where n is the number of samples taken.

The Central Limit Theorem also tells us that if x is not Normally distributed, but our sample size, n is large ($n > 30$), then the sample mean is approximately Normal with mean μ and standard deviation $\sigma/\sqrt{n}$.

A histogram of a typical system's $\overline{\text{AP}}$ is shown in Figure 1. It shows that the sample mean is approximately Normal. This is also the case for all other systems. Therefore, to begin, we will assume that a system's $\overline{\text{AP}}$ follows a Normal distribution, where each system distribution is characterised by its mean μ_{AP} and standard deviation σ_{AP}.

We will also assume that we know each systems standard deviation (σ_{AP}). This is not a useful assumption in practice, but it will allow us to investigate if our assumption of Normality is valid.

Given that $\overline{\text{AP}}$ is Normal for each system, we can compute the confidence interval of μ_{AP} using:

$$\mu_{\text{AP}} \in \overline{\text{AP}} \pm Z_{\alpha/2}\sigma_{\text{AP}}/\sqrt{n} \qquad (1)$$

where $\alpha \in [0, 1]$ is the probability of a Type I error, the level of confidence is $100(1-\alpha)\%$, and $Z_{\alpha/2}$ is the $\alpha/2$ quantile of the Standard Normal distribution (meaning that $100(1-\alpha)\%$ of the Standard Normal distribution lies between $-Z_{\alpha/2}$ and $Z_{\alpha/2}$).

Our first experiments examines the Type I error (α) of the confidence interval. Using a set of 110 system scores, we compute an estimate of μ_{AP} and σ_{AP} using the AP results from all 249 queries. By taking a random sample of $n = 5$ AP scores for a particular system, we are able to compute the confidence interval of μ_{AP} and compare it to the our computed value of μ_{AP}. If μ_{AP} does not lie within the confidence interval, a Type I error has occurred. The value of α provided in the confidence interval calculations is the expected Type I error. Therefore, repeated experiments should show the Type I error of the confidence interval to be equal to α. For our experiment, we computed 1000 confidence intervals for each system, from random samples of $n = 5$ AP scores. The results are presented in Table 1

Table 1: The actual Type I error produced when computing μ_{AP} confidence intervals using knowledge of σ_{AP}, given α. The mean, standard deviation and maximum across all systems are computed from 1000 confidence intervals using $n = 5$ for each system.

α	Type I error		
	Mean	SD	Max
0.050	0.040	0.005	0.051
0.100	0.088	0.009	0.109
0.150	0.139	0.012	0.161
0.200	0.191	0.014	0.208
0.250	0.242	0.014	0.269
0.300	0.295	0.013	0.320
0.350	0.348	0.011	0.376
0.400	0.400	0.010	0.424
0.450	0.452	0.010	0.482
0.500	0.502	0.010	0.533

If the system sample mean distributions are Normal, we would expect to see that the Type I error from all systems be close to the given α. The results show that the Type I error across all systems is close to the value of α implying that the confidence interval being used is correct. Similar results are obtained when using other values of n. These results imply that our assumption that $\overline{\text{AP}}$ is Normal is valid.

4.2 Confidence when σ_{AP} is unknown

In the previous section we assumed that σ_{AP} is known, which would not be the case when estimating a confidence interval, but it allowed us to examine the assumption that $\overline{\text{AP}}$ followed an approximate Normal distribution.

In this section, we assume that σ_{AP} is unknown and therefore we must approximate its value with our sample standard deviation s_{AP}. If AP follows a Normal distribution, then Cochran's theorem [1] provides us with:

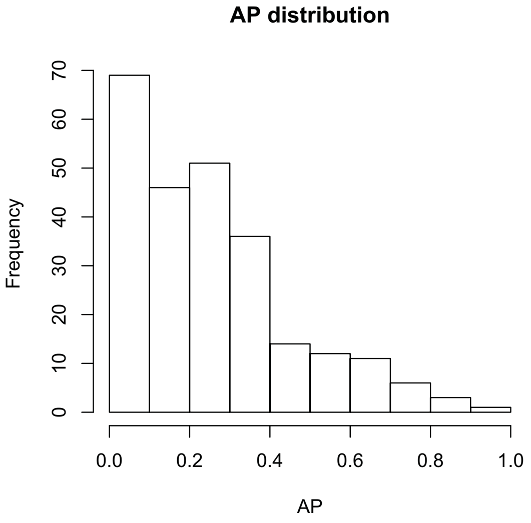


Figure 2: Distribution of a randomly sampled system's scores (AP).

$$\frac{(n-1)s^2_{\text{AP}}}{\sigma^2_{\text{AP}}} \sim \chi^2_{n-1} \qquad (2)$$

where $\sim$ infers equality of distributions and χ^2_{n-1} is the Chi-squared distribution with $n-1$ degrees of freedom.

Figure 2 shows a typical system AP distribution, which does not look Normal, which may infer that the relationship in Cochran's theorem is not valid. The Q-Q plot in Figure 3 shows that the relationship in Cochran's theorem is valid except at the higher end of the scale, implying that the χ^2 distribution has a longer tail than the variance ratio ($(n-1)s^2_{\text{AP}}/\sigma^2_{\text{AP}}$). This implies that score samples with high standard deviation will provide an under estimate of the confidence interval.

By estimating σ_{AP} with s_{AP} using the relationship in equation 2, we arrive at the confidence interval relationship:

$$\mu_{\text{AP}} \in \overline{\text{AP}} \pm t_{\alpha/2,n-1}s_{\text{AP}}/\sqrt{n} \qquad (3)$$

where $t_{\alpha/2,n-1}$ is the $\alpha/2$ quantile of the Student's t distribution with $n-1$ degrees of freedom (meaning that $100(1-\alpha)\%$ of the t distribution lies between $-t_{\alpha/2,n-1}$ and $t_{\alpha/2,n-1}$).

Table 2 shows the results from computing 1000 confidence intervals for each system from samples of $n = 5$ scores, using equation 3. Note that if the system scores were Normally distributed, the computed Type I error would be similar to the given α. We can see that The mean Type I error is greater than α implying that we are under estimating the confidence interval width. The column providing the maximum Type I error shows a large underestimate of the confidence interval. This can be explained from our observation of the Q-Q plot

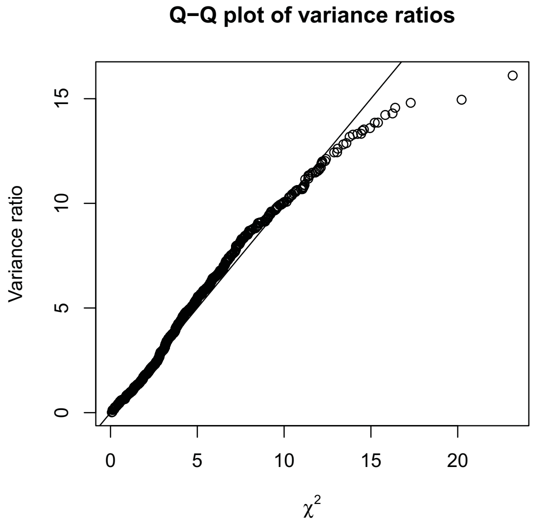


Figure 3: The Q-Q plot of the χ^2_{n-1} distribution against the $(n-1)s^2_{\text{AP}}/\sigma^2_{\text{AP}}$ distribution, for $n = 5$.

in Figure 3, showing that the samples that had larger variance do not follow the χ^2_{n-1} distribution.

Table 2: The actual Type I error produced when computing μ_{AP} confidence intervals using s_{AP}, given α. The mean, standard deviation and maximum across all systems are computed from 1000 confidence intervals using $n = 5$ for each system.

α	Type I error		
	Mean	SD	Max
0.05	0.082	0.027	0.255
0.10	0.133	0.027	0.299
0.15	0.179	0.026	0.340
0.20	0.224	0.024	0.377
0.25	0.269	0.021	0.407
0.30	0.315	0.020	0.440
0.35	0.362	0.018	0.480
0.40	0.410	0.018	0.520
0.45	0.458	0.018	0.559
0.50	0.504	0.018	0.602

We now have the problem that we are unable to obtain a good estimate of the score population standard deviation σ_{AP} and hence unable to obtain an accurate confidence interval for μ_{AP} from a sample of scores. To proceed, we must either obtain the distribution of $(n-1)s^2_{\text{AP}}/\sigma^2_{\text{AP}}$, or find a mapping that provides us with Normally distributed AP. In the next section, we will examine the latter using score standardisation.

5 Standardised AP

Score standardisation was introduced as a method of allowing cross collection comparison of system scores [4]. In this section, we will examine the effect of standardisation on the distribution of AP and its effect on confidence interval estimations.

Standardised AP is defined as:

$$\text{sAP}_q = \frac{\text{AP}_q - \overline{\text{AP}}_q}{s_{\text{AP},q}}$$

where sAP_q is the standardised AP for a given system on query q, AP_q is the Average Precision for the given system on query q, $\overline{\text{AP}}_q$ is the mean AP across a set of systems for query q, and $s_{\text{AP},q}$ is the standard deviation across a set of systems for query q. From this definition, we can see that standardisation is highly dependent on the set of systems (from which $\overline{\text{AP}}_q$ and $s_{\text{AP},q}$ are computed). Therefore, we will begin the investigation using all systems to perform the standardisation and finish by examining the effect of using a small sample to perform standardisation.

We will use the following notation:

- sAP is the standardised Average Precision from a sample query for a given system,
- $\overline{\text{sAP}}$ is the sample mean standardised Average Precision for a given system from a sample of n queries,
- μ_{sAP} is the population mean standardised Average Precision for a given system,
- s_{sAP} is the sample standard deviation standardised Average Precision for given system from a sample of n queries,
- σ_{sAP} is the population standard deviation standardsed Average Precision for a given system,
- $\sigma_{\overline{\text{sAP}}}$ is the population standard deviation of the sample mean standardised Average Precision for a given system.

where μ_{sAP} and σ_{sAP} are estimated using all 249 queries.

5.1 Standardisation using all systems

In this section we will use all 110 systems to compute the mean and standard deviation of each query to perform standardisation. Note that when performing retrieval experiments, it would be unlikely to have evaluated 110 systems on the set of queries being evaluated. Therefore, this section is similar to a 'best case' analysis. We also present the confidence intervals for when σ_{sAP} is known and unknown to identify where any problems in our assumptions lie.

5.1.1 Confidence when σ_{sAP} is known

By performing the standardisation, we obtain a sAP score for each query. To establish the confidence interval for μ_{sAP}, we must deduce the distribution of $\overline{\text{sAP}}$. A histogram of the distribution of a system's $\overline{\text{sAP}}$ is shown in Figure 4. We can see that the particular system sample mean sAP is approximately Normal. If we

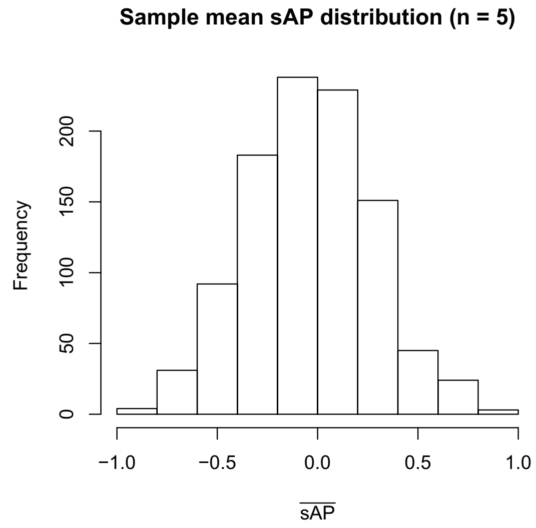


Figure 4: Distribution of a randomly sampled system's sample mean ($\overline{\text{sAP}}$) using $n = 5$.

examine the $\overline{\text{AP}}$ distribution in Figure 1, we find that the $\overline{\text{sAP}}$ distribution is less skewed giving it a more Normal appearance. This Normality implies that we should obtain accurate confidence intervals when the system population standard deviation σ_{sAP} is known.

To compute the accuracy of the confidence interval estimates when σ_{sAP} is known, we used equation 1 and replaced AP with sAP. Samples of size $n = 5$ were used to compute the confidence interval and compared to μ_{sAP}. If μ_{sAP} was not in the confidence interval, a Type I error occurred. This was repeated 1000 times for each system. The probability of a Type I error is listed in Table 3. Table 3 reports the mean, standard deviation and maximum probability of a Type I error across all systems. The table shows mean and maximum values similar to the associated values of α, and small standard deviation. This implies that the confidence intervals produced are accurate.

5.1.2 Confidence when σ_{sAP} is unknown

We have found that the Normal distribution is a good approximation for the distribution of $\overline{\text{sAP}}$. In this section we will examine if we can approximate σ_{sAP} using s_{sAP} and Cochran's theorem (equation 2).

Cochran's theorem is valid under the assumption that the data follows a Normal distribution. The histogram of a sample system's sAP in Figure 5 shows that sAP is approximately Normal. To examine if this approximation is close, we have also examined the Q-Q plot of the variance ratio on the left hand side of equation 2 compared to the χ^2 distribution on the right hand side of equation 2. The plot (given in Figure 6) shows that the two distributions are approximately equal, suggesting that we are able to use s_{sAP} to approximate σ_{sAP}.

Table 3: The actual Type I error produced when computing μ_{sAP} confidence intervals using σ_{sAP}, given α. The mean, standard deviation and maximum across all systems are computed from 1000 confidence intervals using $n = 5$ for each system.

α	Type I error		
	Mean	SD	Max
0.050	0.046	0.006	0.062
0.100	0.094	0.009	0.118
0.150	0.142	0.012	0.175
0.200	0.192	0.014	0.221
0.250	0.243	0.015	0.273
0.300	0.292	0.016	0.324
0.350	0.342	0.016	0.376
0.400	0.392	0.016	0.424
0.450	0.443	0.017	0.474
0.500	0.493	0.017	0.533

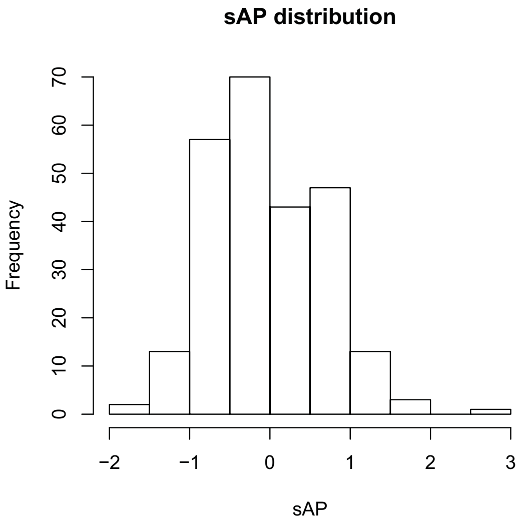


Figure 5: Distribution of a randomly sampled system's scores (sAP).

The confidence interval is computed using equation 3, where we replace all occurrences of AP with sAP. We investigated the accuracy of the confidence interval by computing the confidence interval for 1000 samples of $n = 5$ queries for each system for varying levels of α. Statistics of the Type I error are reported in Table 4. We can see that the expected Type I error (mean) is close to the given α, showing that the confidence interval is accurate.

5.2 Standardisation using a few systems

We mentioned in the previous section that it is unlikely that we would have the results from 110 systems to perform standardisation. Therefore in this section, we will examine the effect of using a random sample of five systems to perform standardisation.

To test the accuracy of our confidence intervals, we ran the same Type I error experiment from Section 5.1.2 except we used only five randomly sampled systems

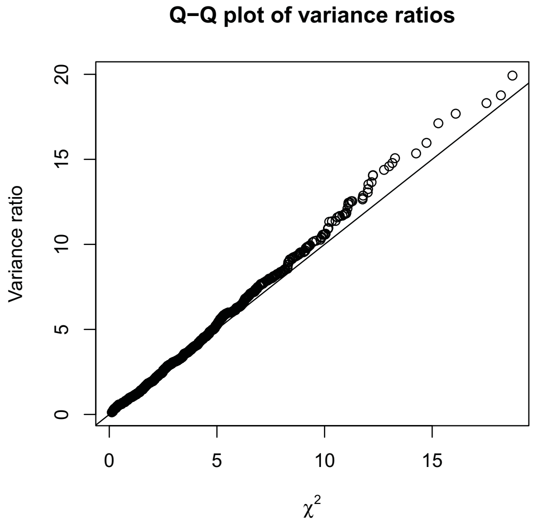


Figure 6: The Q-Q plot of the χ^2_{n-1} distribution against the $(n-1)s^2_{\text{sAP}}/\sigma^2_{\text{sAP}}$ distribution, for $n = 5$.

Table 4: The actual Type I error produced when computing μ_{sAP} confidence intervals using s_{sAP}, given α. The mean, standard deviation and maximum across all systems are computed from 1000 confidence intervals using $n = 5$ for each system.

α	Type I error		
	Mean	SD	Max
0.050	0.050	0.010	0.090
0.100	0.097	0.011	0.143
0.150	0.146	0.012	0.195
0.200	0.195	0.014	0.248
0.250	0.245	0.016	0.294
0.300	0.295	0.017	0.334
0.350	0.346	0.018	0.383
0.400	0.397	0.019	0.441
0.450	0.448	0.020	0.508
0.500	0.499	0.020	0.559

for standardisation. The results from the experiment are shown in Table 5. We can see that the expected (mean) Type I error follows α closely. In comparison to Table 4, we can see that the difference between α and the expected Type I error has increased. We can also see that the variance has increased. Therefore, reducing the number of standardisation systems has slightly decreased the accuracy of the confidence intervals, but they are more accurate than when using AP.

Note that the population mean and standard deviation are dependent on the standardising systems chosen, therefore, we cannot compare system confidence intervals when the systems have used different standardisation systems.

Table 5: The actual Type I error produced when computing μ_{sAP} confidence intervals using s_{sAP} and five randomly sampled standardisation systems, given α. The mean, standard deviation and maximum across all systems are computed from 1000 confidence intervals using $n = 5$ for each system.

α	Type I error		
	Mean	SD	Max
0.050	0.062	0.017	0.157
0.100	0.114	0.021	0.212
0.150	0.166	0.024	0.270
0.200	0.217	0.025	0.312
0.250	0.270	0.026	0.358
0.300	0.324	0.025	0.408
0.350	0.379	0.025	0.468
0.400	0.435	0.025	0.520
0.450	0.491	0.026	0.579
0.500	0.543	0.025	0.634

Table 6: The change in Type I error as n increases, where $\alpha = 0.05$ and σ_{AP} and σ_{sAP} are unknown.

n	2	5	10	20	50
AP error	0.081	0.083	0.066	0.053	0.029
sAP error	0.050	0.062	0.068	0.068	0.054

6 Examination of Confidence Intervals

Confidence intervals are useful for identifying the likely region in which the system population mean exists. As the interval grows, the utility decreases. E.g. we could provide a 100% confidence interval for μ_{AP} as $[0, 1]$. This is accurate, but does not provide us with any information since the confidence interval covers the domain of AP. In this section, we will examine the confidence intervals that were computed in the previous sections.

For the previous experiments, we have used $n = 5$ queries. We now examine the accuracy of our confidence intervals as the number of queries n increases (when σ_{AP} and σ_{sAP} are unknown and using five standard systems). We can see in Table 6 that the Type I error for μ_{sAP} is stable, while the Type I error for μ_{AP} reduces as n increases. This can be explained by the variance ratio ($(n-1)s^2_{\text{sAP}}/\sigma^2_{\text{sAP}}$) following a χ^2_{n-1} distribution. The t distribution, used to compute the confidence interval, is constructed by combining the uncertainty in σ_{sAP} given by the χ^2_{n-1} distribution with the Standard Normal distribution in equation 1. Since the variance ratio of sAP approximately follows a χ^2_{n-1} distribution, the t distribution compensates for the change in n. The variance ratio of AP does not follow a χ^2_{n-1} distribution, therefore the t distribution poorly compensates for n.

The confidence interval equation (shown in Equation 3) is centered at the sample mean $\overline{\text{sAP}}$ and its width is dependent on the sample standard deviation s_{AP}, the error rate α and the number of samples n. The sam-

Table 7: The change confidence interval (CI) width for sAP as n increases, where $\alpha = 0.05$ and σ_{sAP} is unknown.

n	2	5	10	20	50
CI width	20.217	3.456	2.172	1.518	0.981

ple mean and standard deviation of the system score are under our control in an experimental environment, since they are the responses we are examining. In all information retrieval experiments, we have direct control over n, the number of queries used in the retrieval experiment, and α.

By increasing α we decrease the confidence interval, but we also decrease the confidence of the confidence interval. By increasing n, we decrease the confidence interval, but increasing n involves using a larger query set, which (if not already available) involves building the relevance judgements for the new queries. If queries are available, they should be used to increase n and obtain a narrower confidence interval that will be more useful for identifying the location of μ_{sAP}.

The standard deviation of the set of all sAP scores across all 110 systems, each using 1000 different randomly sampled sets of five standardisation systems is 2.436. Therefore, Table 7 shows that we need to use $n = 10$ queries to get an expected confidence interval width that is less than the standard deviation of the samples sAP. This is not a benchmark, but simply an indicator to compare the size the confidence intervals relative to the data.

Table 8: Type I error for the μ_{sAP} confidence interval (with unknown σ_{sAP}) on the 40 systems from TREC-3, using 1000 samples of $n = 5$ queries for each system and five standardisation systems.

α	Type I error		
	Mean	SD	Max
0.050	0.055	0.021	0.142
0.100	0.104	0.024	0.204
0.150	0.153	0.025	0.249
0.200	0.203	0.026	0.293
0.250	0.253	0.024	0.340
0.300	0.303	0.026	0.379
0.350	0.355	0.027	0.421
0.400	0.409	0.028	0.471
0.450	0.462	0.028	0.525
0.500	0.514	0.027	0.572

To test the generalisation of our results, we examined the accuracy of the confidence interval method from Section 5.2 on results from TREC-3. The results in Table 8 show an expected Type I error close to the value of α, with small standard deviation. This implies that this method of computing confidence intervals does generalise.

To report results so that others are able to compare new systems, we need to report the sample mean and sample standard deviation of Average Precision, and the number of queries used. We also need to report which systems were used to perform standardisation. Note that these systems must be freely available systems. If others do not have access to the set of standardisation systems, the confidence intervals cannot be compared. Once these items are reported, others can compute comparative confidence intervals without access to our system, queries or relevance judgements.

7 Conclusion

Current forms of information Retrieval report a sample mean and the confidence obtained using paired hypothesis tests. These values provide the reader with knowledge of which system is more accurate from those taking part in the experiment. Unfortunately, these values do not provide the reader with any means of comparing systems found published in different articles. We can use the system's sample mean as an estimate of the system's population mean (expected value), but the reader has no knowledge of the accuracy of this estimate.

To compare systems across publications, we would need some indication of the systems population parameters. From sample statistics, we are able to compute a confidence interval of the population mean given a certain level of confidence, as long as the sample follows a known distribution function.

In this article, we investigated the distribution of Average Precision for a set of systems and examined if we could construct accurate confidence intervals of the population mean Average Precision from a system's sample statistics.

We found that accurate confidence interval could be constructed when score standardisation was applied. Our analysis showed that we could obtain highly accurate confidence intervals for any number of sample queries while using only five standardisation systems.

References

[1] W. G. Cochran. The distribution of quadratic forms in a normal system, with applications to the analysis of covariance. *Mathematical Proceedings of the Cambridge Philosophical Society*, Volume 30, pages 178–191, 1934.

[2] William Feller. *An Introduction to Probability Theory and Its Applications*. Wiley publications in statistics. John Wiley and Sons, Inc., New York, 2nd edition, 1975.

[3] Sabrina Keenan, Alan F. Smeaton and Gary Keogh. The effect of pool depth on system evaluation in TREC. *Journal of the American Society for Information Science and Technology*, Volume 52, Number 7, pages 570–574, 2001.

[4] William Webber, Alistair Moffat and Justin Zobel. Score standardization for inter-collection comparison of retrieval systems. In *SIGIR '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 51–58, New York, NY, USA, 2008. ACM.

A Meta-Analysis of the Effects of Search Experience on Search Performance in Terms of the Recall Measure in Controlled IR User Experiments

Ying-Hsang Liu
School of Information Studies
Charles Sturt University
Wagga Wagga NSW 2678 Australia
yingliu@csu.edu.au

Abstract

This paper reports a meta-analysis of the effects of search experience on search performance in terms of the recall measure in controlled IR user experiments. More specifically, this study was designed to answer the research question: how large is the average effect size in the set of studies included in the meta-analysis? Search experience, a manifestation of users' search skills accumulated through their interactions with IR systems over time, has been identified as an important research variable in user search behaviours. The participants included in primary studies were end-users or intermediaries recruited for IR user experiments. The results of the meta-analysis (N = 8) using a fixed-effects model showed that search experience has an overall positive effect on the recall measure (weighted mean correlation coefficient r = 0.04, 95% confidence interval was -0.01 to 0.09). Our findings may provide implications for designing adaptive or personalized IR systems that take into account the contextual information at the user and interactional levels.

Keywords

Information Retrieval, User Studies Involving Documents

1 Introduction

Search experience has been identified as one of the key user characteristics that affect search performance in information retrieval (IR) user experiments (see e.g., [13, 14]). While the search experience as an important research variable has been operationalized in various ways for research purposes, search experience in general is a manifestation of users' search skills accumulated through their interactions with IR systems over time.

Proceedings of the 15th Australasian Document Computing Symposium, Melbourne, Australia, 10 December 2010.

Previous studies that were conducted in the 1980s and early 1990s revealed that end-users usually had limited experiences searching online bibliographic databases, because online searching was very expensive and professional librarians usually conducted the search on behalf of users. Here search experience usually referred to whether searchers have had extensive use of online databases and whether they were proficient in the system features, such as search commands or indexing thesauri.

For example, the search experience was measured by the total number of searching sessions in a longitudinal study of medical students' use of MEDLINE [16]. Several studies that examine the effect of search experience on searching behaviour have used the total time spent using a particular online database or DIALOG system as a measure of different levels of search experience [6, 11]. In other studies that investigated whether search success depends on searchers' individual characteristics, the search experience was determined by formal training in online database searching [1, 20].

More recent studies tend to assess whether the search experience in a specific type of information retrieval system can be transferred to another. For instance, since one of the primary objectives was to investigate the effect of online database search experience on Web search performance, researchers used the duration and frequency of using online databases to measure undergraduate students' search experience [15]. Because of the similar system features in Boolean logic, researchers used the frequency of online public access catalogue as a measure of undergraduate students' search experience in a Boolean-based online database [23].

Overall, the participants included in these studies were end-users or intermediaries who conducted searches on behalf of users, and their different levels of search experiences were measured by formal training in online database searching or various kinds of indicators of their exposure to IR systems.

The choice of performance measures of precision and recall has been widely used in evaluating the effectiveness of automatic indexing techniques, in part because researchers can test the performance of different retrieval techniques in a laboratory environment. While user-oriented measures, such as user satisfaction and utility, have been proposed as measures of user search performance, the precision and recall measures still dominate IR experimentation research. We particularly considered the recall measure as dependent variable since it has also been extensively used in previous IR user experiments, and several researchers hypothesized that search experience is correlated with the search outcome in terms of the recall measure [6, 11, 21].

Our review of related studies have focused on controlled IR user experiments because they have high levels of internal validity and allow us to examine the subtle effects of individual differences on search performance in laboratory settings.

Despite different measurement in these user studies, the study of the impact of search experience on search performance has had a growing body of research (See [14] for a recent review). One of the outstanding questions is whether searchers' individual characteristics, such as search experience, are correlated with the measures of search performance? If the answer is yes, how can we estimate the effect of search experience on search performance?

To advance our understanding of the impact of search experience on search performance, this study was designed to collect, analyse and synthesize the empirical findings from controlled IR user experiments. The results will not only help us better understand the impact of individual differences on search performance, but also provide implications for designing adaptive or personalized IR systems that take into account the contextual information at the user and interactional levels.

We conducted a quantitative review of empirical studies by comparing and synthesizing separate results from the research literature. The technique of meta-analysis allowed us to synthesize the research results and determine the relationships between variables. More specifically, our research question is: *how large is the average effect size in the set of studies included in the meta-analysis?* In view of previous research, we formulate the following research hypothesis: *Experienced searchers will perform better than novice searchers in terms of the recall measure*.

2 Method

To collect the empirical controlled IR user experimental studies, we conducted a comprehensive search of Web of Science databases, specifically Social Science Citation Index SSCI) and Science Citation Index (SCI) in August 2008. By using the citation pearl growing search strategy [8], which was designed to use citation relationships to find relevant articles, we were able to systematically collect eligible studies for inclusion in the review.

Originally we had four pearl (or seed) articles drawing from the researcher's knowledge: Pao and her colleagues [16], Howard [11], Fenichel [6] and Sutcliffe, Ennis and Watkinson [21]. The reviewed articles in the dataset of [14] were also included as seed articles because they contain some potentially relevant studies. Using the cited reference function, with particular attention to name variants and inconsistencies of citations, our searches yielded a total of 537 unique references. The study eligibility criteria were controlled IR user experiments that involved the variables of search experience and search performance in terms of the recall measure. The researcher examined the title and abstract of each bibliographic record. Full-text of the articles were consulted if the study has a good chance of fulfilling the above mentioned eligibility criteria. A total of 104 full-text articles were examined. Our selection only resulted in two definitely relevant articles; another three was collected from an examination of cited references in the articles.

For descriptive purposes, each study was coded by searcher characteristics, sample size, IR system used, test collection, search task and outcome measure (See Appendix 1). Note that most studies used Boolean-based IR systems for experimental purposes, and the experimentation of retrieval techniques was not the primary objectives.

To measure the strength of the linear relationship between two variables, i.e., search experience and search performance, we selected correlation coefficient *r*. The effect size of these studies was transformed into raw correlation coefficients because the search experience variable was measured in a wide variety of ways and the outcome variable of recall was applied in different ways (See Appendiex 1). In these situations regression coeffcents are not directly comparable across all the studies, while correlation coefficients can be compared [19].

Correlation coefficients of included studies were calculated based on the experimental design, sample size and details of reported statistics, using the formula in Borenstein [2] and the functions in R statistical software [5, 18]. In general, correlation coefficients can be easily computed if the report provides *F* value for one-way ANOVA in comparing two groups. When the *F* value was not available and the raw data was presented in the report, a one-way ANOVA was conducted (See [21]). For repeated measure study, such as [9], we followed the procedure in [19]. In other cases where the *F* value or *p*-value of insignificant results was not reported, the effect size was replaced with a value of zero [17], including studies of Fenichel [6], Howard [11] and Pao et al. [16].

To estimate the magnitude of search experience on search performance in terms of the recall measure, we fit the data into a fixed-effects model [12]. We assumed that there is true effect of search experience across all the studies. After deriving the raw correlation coefficient, we performed the Fisher's *r*-to-*z* transformation for normalization. The meta-analysis with a fixed-effects model was conducted using metafor package [18, 24].

3 Results

This study was designed to integrate studies that investigated the impact of search experience on search performance in terms of the recall measure in controlled IR user experiments. After the systematic collection and examination of potentially relevant articles, our corpus consists of 9 studies.

To test whether the true effect is homogeneous, a test for homogeneity revealed that homogeneity of correlations is rejected (Q = 68.09, *df* = 8, *p* < .0001). We then calculated leave-one-out diagnostics that indicates the effect of deleting one case on the fitted model [7, 24]. The results indicated that the amount of heterogeneity is significantly reduced by removing Hersh and Hickam's [9] study (Q= 7.52, *df* = 7, *p* = 0.38). Further examination of this study showed that methodologically it is different from other included studies because of the use of replicated searches for comparing search performance between librarians and physicians. Therefore, our final results were based on a corpus of 8 studies, excluding the out-lying case.

To gauge the size of homogeneity, the I^2 statistic was calculated [10]. The I^2 = 6.9% was considered small heterogeneity, suggesting that only about 7% of variation in effect sizes is due to heterogeneity

Results of the meta-analysis (*N* = 8) showed that search experience has an overall positive effect on the recall measure (weighted mean correlation coefficient *r* = 0.04, 95% confidence interval was -0.01 to 0.09), as shown in Figure 1. The figure indicated that only Chen's [4] study has demonstrated significantly positive effect of search experience, as the lower bound of confidence interval (CI) does not cross the vertical line, with zero Fisher's *z* transformed correlation coefficient. The sizes of rectangular represent sample size for each study, whereas the diamond summarizes the averaged effect size. Because the average effect size *r* = 0.04 and 95% CI was between -0.01 and 0.09, our hypothesis that experienced searchers will perform better than novice searchers in terms of the recall measure was not supported.

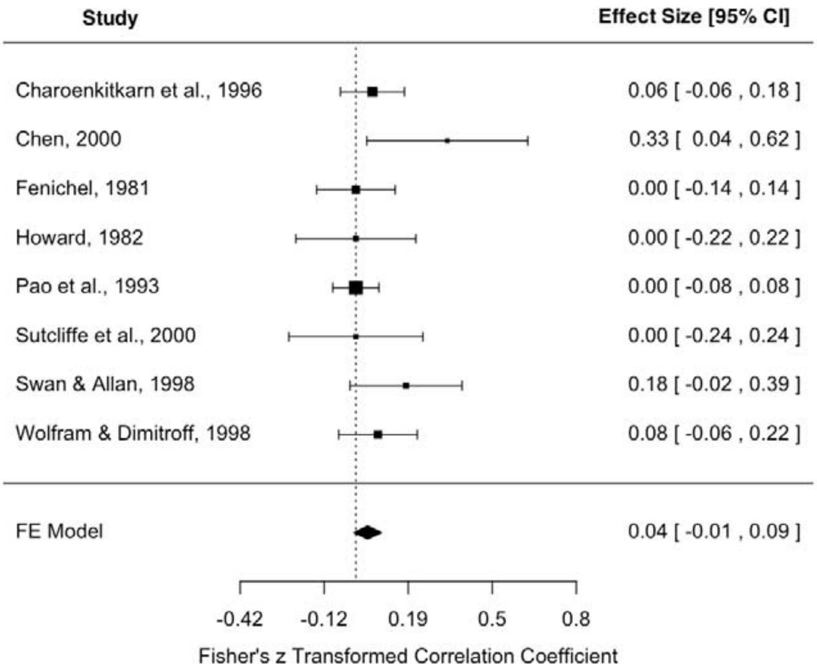


Figure 1. Forest plot of the effect of search experience on search performance in terms of the recall measure in controlled IR user experiments.

4 Conclusion

This meta-analytic study was designed to estimate the effect of search experience on search performance in terms of the recall measure in controlled IR user experiments. Our results ($N = 8$) indicated that search experience overall has an overall positive effect on the recall measure (weighted mean correlation coefficient $r = 0.04$, 95% confidence interval was -0.01 to 0.09). However, the hypothesis that experienced searchers will perform better than novice searchers in terms of the recall measure was not supported.

5 References

[1] Bellardo, T. An investigation of online searcher traits and their relationship to search outcome. *Journal of the American Society for Information Science*, 36, 4 1985), 241-250.

[2] Borenstein, M. *Effect sizes for continuous data*. Russell Sage Foundation, New York, 2009.

[3] Charoenkitkarn, N., Chignell, M. and Golovchinsky, G. Is recall relevant? An analysis of how user interface conditions affect strategies and performance in large scale text retrieval. *Proceedings of the Fourth Text REtrieval Conference (TREC-4)*1997), 211-232.

[4] Chen, C. Individual differences in a spatial-semantic virtual environment. *Journal of the American Society for Information Science*, 51, 6 2000), 529-542.

[5] Del Re, A. *compute.es: Compute effect sizes*. R package, 2010.

[6] Fenichel, C. H. Online searching: Measures that discriminate among users with different types of experiences. *Journal of the American Society for Information Science*, 32, 1 (Jan 1981), 23-32.

[7] Greenhouse, J. B. *Sensitivity analysis and diagnostics*. Russell Ssage Foundation, New York, 2009.

[8] Harter, S. P. *Online information retrieval: Concepts, principles, and techniques*. Academic Press, Orlando, FL, 1986.

[9] Hersh, W. and Hickam, D. Use of a multi-application computer workstation in a clinical setting. *Bulletin of the Medical Library Association*, 82, 4 1994), 382-389.

[10] Higgins, J. P. T., Thompson, S. G., Deeks, J. J. and Altman, D. G. Measuring inconsistency in meta-analyses. *Br. Med. J.*, 327, 7414 (Sep 2003), 557-560.

[11] Howard, H. Measures that discriminate among online searchers with different training and experience. *Online Review*, 6, 4 (Aug 1982), 315-327.

[12] Konstantopoulos, S. and Hedges, L. V. *Analyzing effect sizes: Fixed effects models*. Russell Sage Foundation, New York, 2009.

[13] Meadow, C. T., Marchionini, G. and Cherry, J. M. Speculations on the measurement and use of user characteristics in information retrieval experimentation. *Canadian Journal of Information and Library Science*, 19, 4 1994), 1-22.

[14] Moore, J. L., Erdelez, S. and Wu, H. The search experience variable in information behavior research. *Journal of the American Society for Information Science and Technology*, 58, 10 2007), 1529-1546.

[15] Palmquist, R. A. and Kim, K. S. Cognitive style and on-line database search experience as predictors of Web search performance. *Journal of the American Society for Information Science*, 51, 6 (Apr 2000), 558-566.

[16] Pao, M. L., Grefsheim, S. F., Barclay, M. L., Woolliscroft, J. O., McQuillan, M. and Shipman, B. L. Factors affecting students use of MEDLINE. *Computers and Biomedical Research*, 26, 6 (Dec 1993), 541-555.

[17] Pigott, T. D. *Hadling missing data*. Russell Sage Foundation, New York, 2009.

[18] R Development Core Team *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, 2010.

[19] Rosenthal, R., Rosnow, R. L. and Rubin, D. B. *Contrasts and effect sizes in behavioral research: A correlational approach*. Cambridge University Press, New York, 2000.

[20] Saracevic, T., Kantor, P., Chamis, A. Y. and Trivison, D. A study of information seeking and retrieving: I. Background and methodology. *Journal of the American Society for Information Science*, 39, 3 1988), 161-176.

[21] Sutcliffe, A. G., Ennis, M. and Watkinson, S. J. Empirical studies of end-user information searching. *Journal of the American Society for Information Science*, 51, 13 2000), 1211-1231.

[22] Swan, R. C. and Allan, J. Aspect windows, 3-D visualizations, and indirect comparisons of information retrieval systems. In *Proceedings of the 21st ACM SIGIR Conference* (Melbourne, Australia, 1998). ACM, New York.

[23] Vakkari, P., Pennanen, M. and Serola, S. Changes of search terms and tactics while writing a research proposal: A longitudinal case study. *Information Processing & Management*, 39, 3 2003), 445-465.

[24] Viechtbauer, W. Conducting meta-analyses in R with the metafor package. Journal of Statistical Software, 36, 3 2010), 1-48.

[25] Wolfram, D., Volz, A. and Dimitroff, A. The effect of linkage structure on retrieval performance in a hypertext-based bibliographic retrieval system. *Information Processing & Management*, 32, 5 1996), 529-541.

Appendix 1. Descriptive analysis of the effect of search experience on search performance

Study	User	Sample Size *N*	IR System	Collection	Search Task	Outcome Measure
1. Charoenkitkarn et al., 1996 [3]	Most experienced searchers had extensive online searching experiences, and performed searches on a daily basis.	36 searchers × 8 topics = 288 searches	Information exploration system, with different search interface conditions	TREC-3 test documents	Find answers to search topics; 8 search topics from TREC-3	Standard recall
2. Chen, 2000 [4]	The average online search experience was 5 years.	12 searchers × 4 topics = 48 searches	Information visualization system, with textual and spatial search interfaces	169 articles from ACM CHI conference proceedings	4 search topics; save relevant articles for each topic	LSI (Latent Semantic Indexing)-based recall scores
3. Fenichel, 1981 [6]	Experienced searchers were regular users of DIALOG and novice searchers were beginning MLIS students.	48 searchers × 4 topics = 192 searches	ERIC ONTAP on the DIALOG system, command line interface	35,000 bibliographic references, about 12% of the ERIC database	4 search topics	Standard recall
4. Hersh & Hickam, 1994 [9]	Experienced searchers in comparison were medical reference librarians and physicians	4 times searched × 106 topics = 424 searches	GRATEFUL MED and ELHILL search interfaces	A subset of MEDLINE covering 270 journals over five years	106 search topics	Standard recall
5. Howard, 1982 [11]	Search experience was distinguished by the length, number of frequency of searches, and ERIC use experience	42 searchers × 2 topics = 84 searches	DIALOG system, command line interface	ERIC database	2 search topics	Standard recall
6. Pao et al., 1993 [16]	Medical students' search experience was based on the total number of online sessions	184 searchers × 3 topics = 552 searches	PaperChase search interface	MEDLINE database	3 search topics	Standard recall
7. Sutcliffe et al., 2000 [21]	Medical students' search	17 searchers	WinSPIRS search	MEDLINE database	4 search topics	Standard recall

	experience was based on whether they had some experience using MEDLINE	× 4 topics = 68 searches	interface			
8. Swan & Allan, 1998 [22]	Experienced searchers were librarians who had MLS degrees; Novice searchers were primarily students	16 searchers × 6 topics = 96 searches	Inquery search engine with three different search interfaces	A subset of the TREC collection; articles from the Final Times, approximately 200,000 articles	6 search topics; Identify as many aspects of relevance to a query as one can	Aspectual recall
9. Wolfram & Dimitroff, 1998 [25]	Search experience was based on searchers' self rating	48 searchers × 4 topics = 192 searches	A prototype hypertext system and a Boolean-based system	Approximately 3,000 records from the NTIS database	4 search topics	Standard recall