To Clean or not to Clean: Document Preprocessing and Reproducibility

DWAIPAYAN ROY, Indian Statistical Institute, Kolkata, India
MANDAR MITRA, Indian Statistical Institute, Kolkata, India
DEBASIS GANGULY, IBM Research, Dublin, Ireland

Web document collections such as WT10G, GOV2 and ClueWeb are widely used for text retrieval experiments. Documents in these collections contain a fair amount of non-content-related markup in the form of tags, hyperlinks, etc. Published articles that use these corpora generally do not provide specific details about how this markup information is handled during indexing. However, this question turns out to be important: through experiments, we find that including or excluding metadata in the index can produce significantly different results with standard IR models. More importantly, the effect varies across models and collections. For example, metadata filtering is found to be generally beneficial when using BM25, or language modeling with Dirichlet smoothing, but can significantly reduce retrieval effectiveness if language modeling is used with Jelinek-Mercer smoothing. We also observe that, in general, the performance differences become more noticeable as the amount of metadata in the test collections increase. Given this variability, we believe that the details of document preprocessing are significant from the point of view of reproducibility. In a second set of experiments, we also study the effect of preprocessing on query expansion using RM3. In this case, once again, we find that it is generally better to remove markup before using documents for query expansion.

CCS Concepts: • **Information systems** → *Test collections*; *Retrieval effectiveness*; *Search engine indexing*;

Additional Key Words and Phrases: Reproducibility, Web data, Noise, Relevance Feedback, Metadata Preprocessing, Selecting Indexable Content

ACM Reference Format:
Dwaipayan Roy, Mandar Mitra, and Debasis Ganguly. 2018. To Clean or not to Clean: Document Preprocessing and Reproducibility. *ACM J. Data Inform. Quality* 1, 1, Article 1 (January 2018), 25 pages. https://doi.org/10.1145/3242180

1 INTRODUCTION

Reproducibility of empirical results is a fundamental requirement of disciplines such as Information Retrieval (IR) in which experimentation plays a major role. Experimentation in systems-oriented IR research typically involves two major aspects: an IR system, and one or more test collections. Reproducibility is partly ensured by the widespread use of standardised test collections,

Authors' addresses: Dwaipayan Roy, Indian Statistical Institute, Kolkata, CVPR Unit, 203, B.T. Road, Kolkata, West Bengal, 700108, India, dwaipayan_r@isical.ac.in; Mandar Mitra, Indian Statistical Institute, Kolkata, 203, B.T. Road, Kolkata, West Bengal, 700108, India, mandar@isical.ac.in; Debasis Ganguly, IBM Research, Dublin, Dublin, Ireland, debasis.ganguly1@ie.ibm.com.

1936-1955/2018/1-ART1 $15.00
https://doi.org/10.1145/3242180

experimental protocols and evaluation measures, such as those provided / defined by TREC[1], NTCIR[2], CLEF[3], INEX[4], and FIRE[5]. However, apart from a test collection, experimentation in IR also generally involves a retrieval system. Because most retrieval engines are relatively complex, multi-component, highly configurable systems, precisely reproducing a set of experimental results can be challenging in the absence of a detailed description of the retrieval engine used, and its settings. Such a description should preferably cover at least the following components of the IR system.

(1) Document tokenisation / parsing method used:[6] which parts of a document are tokenised, what characters are regarded as token delimiters, the nature of the tokens themselves (e.g., words, n-grams), which tokens are discarded, if any (e.g., strings consisting of numerals only), etc.
(2) Stopword list used, if any.
(3) Stemming algorithm used, if any.
(4) Additional indexing units (e.g., phrases and named entities) identified, if any, and details of the identification method used.
(5) Details of retrieval model used (e.g., language modeling, smoothing method used, values of parameters, etc.).
(6) Information about other techniques (e.g., reranking, query expansion) that are used on top of a basic, keyword-matching approach to ranked retrieval.

In practice, however, many of the above details are often missing from the system descriptions provided in scholarly articles [29, 49]. In most cases, authors mention only the retrieval model and the engine that was used (particularly when it is an open-source engine such as Indri, Lucene or Terrier) along with some additional information, such as the stopword list used, and the stemming method employed. This has become a standard practice, since the proposed method and/or research findings are the primary focus in published papers, and the publication system does not, as yet, provide a convenient method for recording the mundane details required to reproduce experimental results.[7] These missing details lead to potential reproducibility issues, since the various components of an IR system may, in general, have a significant effect on the overall performance of the system [29, 34, 49].

In this study, we focus in particular on the parsing step (Item 1 above). Our objective, in other words, is to examine how important the details of document parsing are for reproducibility. More specifically, we are interested in the choices made regarding which parts of documents to index, and the impact of these choices on retrieval effectiveness.

A retrospect of recent publications ([3, 9, 25, 27], for example) in leading IR venues shows that papers rarely include complete information about document indexing. Other papers [26, 73, 74] mention whether stopwords were removed and whether words were stemmed, but omit details regarding which parts of documents were indexed.

For document collections consisting primarily of relatively clean, textual content, without meta-information and tags (e.g., newswire collections such as WSJ, AP and SJMN included in Tipster Disks 1–3), the choice is reasonably clear: the entire document should probably be indexed. In contrast, documents in web corpora (e.g. WT10G and ClueWeb) generally contain, in addition

[1] http://trec.nist.gov/
[2] research.nii.ac.jp/ntcir/
[3] www.clef-initiative.eu
[4] http://inex.mmci.uni-saarland.de/about.html
[5] fire.irsi.res.in/
[6] For simplicity, we refer to this step as *parsing* henceforth.
[7] Early TREC proceedings included System Summaries or System Descriptions in an attempt to record some of the above information; however, this practice also seems to have petered out.

to actual information, a significant amount of markup and meta-content (HTML tags along with their options, style specifications, Javascript fragments, etc.), that are used by a browser to visually render the document, but which are unrelated to the information contained in the document. These pieces of a document — informally referred to as *noise* in this article — should probably be discarded during parsing.

Our goal in this study is to confirm the above hypothesis, and thereby to arrive at definitive guidelines with regard to document parsing. We seek to do this by quantifying the impact of noise removal and content selection on system performance, both for baseline retrieval methods, as well as methods involving query expansion. To summarise, the questions that we intend to address are the following.

Question 1. Should documents (particularly documents in Web collections) be cleaned prior to indexing? How much difference does noise-removal make? Does the answer depend on the retrieval model applied?

Question 2. What are the effects of noise removal on query expansion (QE) techniques?

Our experiments show that removing noise during document parsing can indeed significantly affect retrieval effectiveness, *but this effect varies across retrieval models as well as test collections.* In particular, for reasons discussed in Section 6, noise removal *adversely* affects the effectiveness of at least one retrieval model for some collections.

Thus, there does not appear to be a single, obvious answer to the question of whether noise removal is desirable; researchers might reasonably make different choices in different situations with regard to noise removal. For example, even though our findings suggest noise removal is, on the whole, to be preferred, Soboroff [66] provides code that indexes *all* content inside the container tag (<DOC> . . . </DOC>) for each document.

Given the above variations in praxis, and in the absence of definite information about choices made with regard to noise removal, one might have to resort to guesswork in order to reproduce a given set of experimental results. Secondly, an improper choice of parsing method can lead to poor results or weak baselines, which may in turn lead to incorrect conclusions about the benefits of using various retrieval methodologies [5] (such as query expansion, for example). In sum, therefore, we conclude that details about document parsing are important from the perspective of reproducibility.

The rest of this article is organized as follows. In the next section, we discuss previous work in order to provide a context for this study. In Section 3, we describe the method that we used to detect noise in documents. Section 5 describes our experimental setup and the test collections used. Section 6 compares and discusses the performance obtained for different retrieval models with and without noise removal. Section 7 concludes the paper with directions for future work.

2 RELATED WORK

In this section, we first discuss prior work on reproducibility in IR. Subsequently, we review some early work that deals with parsing of documents in general, and of web pages in particular, from an indexing perspective.

2.1 Reproducibility in IR

Given the importance of being able to reproduce experimental results in IR, it is not surprising that the issue of reproducibility has become a specific focus area at a number of recent workshops and conferences [4, 32, 33, 40, 49].

The RIGOR workshop at SIGIR 2015 [4] provided an opportunity to present failed attempts to reproduce previously published results under the same or similar experimental conditions. The

Open-Source IR Reproducibility Challenge [49] was organised as a part of RIGOR. The goal of this challenge was to bring together developers of some commonly used, open-source IR systems in order to create easily reproducible and representative baselines for ad hoc retrieval tasks that can be used as points of comparison by other researchers. The organisers / developers focused on the Gov2 collection, but work on other collections (e.g., ClueWeb) is said to be in progress. The systems' effectiveness scores were seen to vary substantially (although most of the differences were not found to be statistically significant) even among systems that supposedly implemented the same model (e.g., BM25), or shared a common "lineage" (for example, Indri and Galago). The organisers hypothesise that these differences may be attributed to "relatively uninteresting differences" in document pre-processing, tokenization, stemming, and stopword removal. This observation strengthens our motivation: as mentioned in Section 1, we would like to investigate more closely how much of these differences can be accounted for by differences in document pre-processing / parsing methods.

At the same workshop, Di Buccio et al. [29] presented an alternative approach to obtaining reproducible results using open-source IR systems. They first show that even "minor" details (which are invariably omitted from system descriptions) may be the cause for important differences in ranking lists produced by systems. They then prescribe building a detailed taxonomy of components used by open-source IR systems, and carefully documenting the particular configuration (i.e., the specific set of components used, and their settings if any) of a system that leads to a given set of performance scores. This detailed documentation is expected to make it easier for other groups to reproduce a particular set of results.

While RIGOR explicitly focused on IR, another seminar that focused on reproducibility issues across sub-disciplines of Computer Science was held in 2016. The so-called PRIMAD (**P**latform, **R**esearch goal, **I**mplementation, **M**ethod, **A**ctor, and **D**ata) model encapsulating the different aspects of an experiment was developed at the seminar as a tool for understanding reproducibility. A report on this workshop [33] discusses reproducibility challenges in the context of both user and system-oriented IR experimentation. A brief overview of reproducibility challenges and issues in IR can also be found in [31].

Yang and Fang [72] present yet another study on reproducibility, but their focus is on retrieval models, rather than whole systems. They present a comprehensive comparison of the performance of 20 different retrieval functions over 16 standard TREC collections. They also make available a web-based framework called RISE, within which researchers can implement additional retrieval functions. This makes it easy to perform a carefully controlled comparison between new retrieval functions (or new implementations of known functions) and existing ones across many collections. However, when comparing multiple retrieval functions over a given collection, RISE provides the same underlying indexes of the collection and eliminates the impact of document pre-processing methods. Our goal in this study is thus complementary: we would like to study the impact of different document parsing methods on the performance of a given retrieval model.

Ferro and Silvello [34] propose a general, statistical framework for analysing the effect of different components on overall system performance. The specific components studied in [34] are stopword lists, stemmers or n-grams, and IR models; document parsing was not included in the scope of Ferro and Silvello's analysis. Moreover, the experiments in [34] were conducted on the test collections used for the TREC 5–8 ad hoc tasks. As discussed in Section 1, our hypothesis is that noise removal is likely to be more impactful for Web collections.

Apart from baseline retrieval methods, we are also interested in pseudo relevance feedback (PRF) based QE methods (see Question 2 from Section 1). PRF based query expansion was carefully studied at the **R**eliable **I**nformation **A**ccess (RIA) workshop [41]. Seven systems were used, all

using PRF based QE. The impact of both system and topic variability on the behavior of PRF for document retrieval was investigated, and extensive failure analysis was done.

2.2 Parsing of documents and Web pages

The test collections used for the ad hoc task during the early years of TREC consisted of documents that were marked up using SGML. Some participating teams (e.g., Cornell University) constructed a list of the various tags present in documents, and prepared a hand-crafted set of rules regarding which tags signify indexable content, and which tags enclose content that should be discarded. Simple safeguards were put in place to handle syntactic errors (e.g., missing closing tags).

To the best of our knowledge, no such list is available for the Web collections (e.g. WT10G and ClueWeb). Therefore, researchers need to decide for themselves which portions of Web documents are to be indexed, and which portions are to be discarded. The community must have faced this issue during the early years of the TREC Web Track [16, 19, 22, 23, 42], but the research focus was primarily on link analysis and differential weighting for different parts of Web pages, rather than on parsing, per se [22, 23, 42, 47].

Of course, parsing of Web pages and extraction of useful content, particularly in the presence of navigational and advertisement blocks, and syntactic errors, is a challenging task that has been extensively studied [11, 38, 56, 68, 69].

Web documents are also vulnerable to the problem of *spamdexing*,[8] or *keyword stuffing*: keywords added to a document for search engine optimization (SEO) may, in fact, be unrelated to its actual content. A great deal of work has been done on IR in the presence of adversarial information providers, and on detecting spam pages [12, 13, 21, 39, 53, 67]. Some work has also been done on studying the effect of spam (or "content manipulation") on classical retrieval models [57], and using content-based techniques to address this problem [52, 59]. Detecting and eliminating spam is beyond the scope of our investigation, however; we are simply interested in studying how baselines may be affected if noise in Web pages (non-content-related portions that are quite possibly legitimate) is removed prior to document indexing.

In the next section, we present the details of how we removed noise, following which, empirical results are presented showing that there is a relationship between noise removal and the performance of a retrieval engine.

3 INDEXING WEB DOCUMENTS

While processing documents from Web collections, broadly speaking, we have two options: either to index whole documents, or to discard portions that correspond to markup and meta-content prior to indexing. Recall from Section 1 that our hypothesis is that the second option should be the preferred alternative.

As an example supporting this hypothesis, consider Figure 1 which shows document WTX103-B39-174 from the WT10G web collection [42]. This is a judged non-relevant document[9] for topic 503 (*Vikings in Scotland?*) from the TREC 10 Web Track [43]. The actual (visible) content of the corresponding Web page is marked by a rectangle and constitutes only a small fraction of the overall document. The "keywords" section does contain terms related to the visible content (e.g., *Opto Couplers, Opto Switches, Photonics*), but because this section is often a target for keyword stuffing / spamdexing [67], it is questionable whether it should be indexed in the same way as the visible content of the document. The remaining words in the document (mostly HTML tags) are

[8] https://en.wikipedia.org/wiki/Spamdexing

[9] This means that the document was retrieved at a relatively high rank by at least one system participating in the TREC 10 Web Track, justifying its inclusion in the pool.

```
<HTML>
<HEAD>
 <META CONTENT="asWedit 2.0" NAME="Generator">
 <META CONTENT="lorna" NAME="AUTHOR">
 <LINK HREF="mailto:lcampbell@cims.co.uk" REL="made">
 <META CONTENT="Thur 8 Aug 1996" HTTP-EQUIV="created">
 <META CONTENT="Scottish Enterprise, Scotland, Scottish Business, Locate in Scotland, Scottish Trade, Scottish
      Enterprise News, locate, location, investment, Bargraphs, Cameras, CMOS Imaging, Consultancy, Contract
      Research, Diode, Display, Electronics, Fibre Optics, Image Sensing, Imaging, Indicators, Infra-Red, Infra-Red
      Gas Sensors, Laser, LCD, Led, Light sources, Optical, Optics, Opto Couplers, Opto Switches, Optoelectronics,
      Photonics, SLM, Smart Vision, Ultraviolet, Visible." NAME="keywords">

 <TITLE>Scotland's key sectors - optoelectronics</TITLE>

</HEAD>

            <FRAMESET COLS="120,*">
            <FRAMESET ROWS="230,*">
            <FRAME SRC="../upperstrap.html" NORESIZE SCROLLING="auto" NAME="upperstrap">
            <FRAME SRC="./sectors.lowerstrap.html" NORESIZE SCROLLING="auto" NAME="lowerstrap">
            </FRAMESET>
            <FRAME SRC="./optoelectronics.body.html" NORESIZE SCROLLING="auto" NAME="body">

            </FRAMESET>
</HTML>
```

Fig. 1: Document WTX103-B39-174 from the WT10G collection. The visible content is marked by a rectangle. Note that this is a judged non-relevant document.

clearly unrelated to the document content, and are therefore unsuitable as indexing terms. Thus, it makes sense for the document preprocessor to remove this noise prior to indexing the document.

To quantify the impact of noise cleaning on system performance, we created two different indexes for each collection. For the *full* index, the original documents (including all markup and meta-content) were indexed; for the *clean* index, documents were run through a preprocessor that removes noise prior to indexing. In the rest of this section, we present the details of how these two document indexes were created.

3.1 Full Index

For the *full* index (also called the *noisy* index in this article), whole documents were indexed as-is using Lucene 5.3.1. Other than a pre-defined set of stopwords, no words were discarded. We followed the basic guidelines provided in [66][10], with only one minor change. Unlike Soboroff [66], we removed the additional document markup fields added by the collection creators (e.g., <DOC>, <DOCHDR>, <DOCNUM> for the WT10G and GOV2 collections, and the WARC information for the ClueWeb09 collection). These tags were not part of the original web pages. [11] Porter's stemmer is used to stem the words before indexing.

[10]This has also been a suggested choice in a number of workshops [7, 70].

[11] Our preliminary experiments (results not reported in this article) show that such a seemingly unimportant decision (about whether the preamble is indexed or eliminated during preprocessing) can also have a significant impact on performance, particularly for LM-JM based initial retrieval, as well as for LM-JM based RM3. For the full index used in our experiments, we chose to remove the preamble tags, as they are not part of the original document.

```
<HTML>
<HEAD>
<TITLE>Highlands and Islands of Scotland − Virtual Tour − Shetland</TITLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF">
<IMG align=right width="120" height="180" VSPACE=5 HSPACE=5 alt="photo" SRC="shet6.jpg">
<H2>LAND OF THE VIKINGS</H2>
<P>
In the ninth century, the whole western world was rocked by the movement of Norsemen away from their own countries, their longships leaving the fjords for new lands across the sea. Their adventuring and colonisation in time took them to the Holy Land sailing south, Greenland and America heading west.
<P>
Much of northern England and Scotland − Caithness, the Western Isles, Orkney and Shetland − succumbed to these forays, and the Picts gave way to this powerful force that came and stayed. Not just warriors, but farmers and their families, and a new culture.
<P>
From 872 AD, a powerful Viking earldom had been established in neighbouring Orkney, and although actual Scandinavian rule in Shetland was to last until the mid fifteenth century, in reality that influence is still incredibly prevalent. Wherever the Vikings went they took their law and their language, and of Shetland's 50,000 place names most are Norn. Their local parliament was held at Lawting Holm, an islet in Tingwall Loch.
<P>
One spectacular and enduring reminder of Shetland's heritage is to be found at Jarlshof− regarded as one of the most interesting and complex archaeological sites in all Britain. A settlement buried in time, until a storm exposed the masonry of an entire village. Wheelhouses and brochs, hearths and troughs reflecting the way of life of a bygone era.
<P>
Equally enduring are the folklore and the sagas associated with many other legendary sites − the Broch of Mousa, the Loch of Girlsta, Haroldswick on Unst, the 'Bears Bait' on Fetlar, and the beach at Gulberwick, for instance, where two of Earl Rognvald's longships were wrecked en route for the Crusades in 1148. A million other secrets must still be locked away in mounds and under fields, beneath sandy beaches and on the seabed.
<P>
<IMG align=left width="280" height="170" VSPACE=5 HSPACE=5 alt="photo of longship burning" SRC="shet7.jpg">
<I>(photo: Burning the Up Helly Aa longship)</I>
<P>
As a reminder. if one were ever needed, of Shetland's Scandinavian past, every year the midwinter festival, Up Helly.Aa, features a procession of a thousand torch−carrying revellers' s, a squad of Vikings in horned helmets and full regalia, and a longship, dragged through the streets of Lerwick, before its ceremonial burning. There's more than a hint of myth and history in this extraordinary celebration.
<P>
<TABLE WIDTH=100% BORDER CELLPADDING=5 CELLSPACING=1>
<TD WIDTH=25%>
<TD WIDTH=25%>
<TD WIDTH=25% ALIGN=CENTER><A HREF="index.html">Shetland</A>
<TD WIDTH=25% ALIGN=CENTER><A HREF="../../index.html"><I>Gael</I>−net Home Page</A>
</TABLE>
<P>
<TABLE WIDTH=100%>
<TD>Comments and e−mail to <A HREF="mailto:info@gael-net.co.uk">info@gael−net.co.uk</a>
<TD ALIGN=RIGHT>Page updated 05/03/96<!-- by KM -->
</TABLE>
</BODY>
</HTML>
```

Fig. 2: Document WTX075-B17-106 from the WT10G collection. This is a judged relevant document for the query *503 - Vikings in Scotland?*

3.2 Clean Index

To generate the *clean* index, we preprocessed the text of documents to remove the following specific types of "noise" prior to indexing.

(1) **HTML tags along with their attributes:** HTML tags are generally not included in stopword lists. Thus, if they are not removed by the preprocessor, the tags in a document will be indexed as a part of the document's content. Since HTML tags are unlikely to occur in a user query, they might not participate in a query-document match, but they do indirectly influence document ranking by increasing a document's length.
Further, if a method such as the Relevance Model [48] is used for query expansion, these tags may constitute a significant part of the expanded query (as we will see in Section 5) due to the high probability of their co-occurring with query terms in the pseudo-relevant documents.

(2) **URLs and email addresses:** Even though URLs and email addresses typically occur as attributes of tags (e.g., they occur in the HREF attribute of <A> tags, or the SRC attribute of <FRAME> tags), and are therefore included in the first type of noise mentioned above, we consider them separately because the URLs in a document may yield terms that match query terms. However, these terms may or may not be related to the actual content of the document. For example, a URL in the document in Figure 1 contains the term *optoelectronics*, which is related to the visible content of the page. In the document shown in Figure 2, however, an unrelated term *index* occurs in URLs. Thus, matches between query terms and terms extracted from URLs may have either a positive or a negative impact on retrieval effectiveness. In any case, the text of the URLs is usually not explicitly displayed by Web browsers, so the URL strings do not necessarily convey any relevant information to the end user.

In order to identify and remove the types of noise described above from documents, we used *jsoup*[12], an HTML parser for Java. According to the jsoup home page, it is "designed to deal with all varieties of HTML found in the wild; from pristine and validating, to invalid tag-soup; jsoup will create a sensible parse tree." Since several types of syntactic errors — non-standard tags, unbalanced tags, incorrect encoding of HTML files — occur frequently in Web pages (Section 2.2), this feature is important. Further, jsoup provides a number of options to extract parts of HTML documents like URLs, email addresses, visible text, etc. As discussed above, all contents of an HTML page other than the visible text may be considered as noise. We therefore used jsoup to explicitly remove all such noise, and to extract only the visible textual content from web pages in order to construct the clean index.

Table 1 presents some statistics related to the full and clean indexes for standard Web collections. The 'Markup-to-Length Ratio' column is of particular interest. This column shows how much of the original collection is eliminated via noise removal. To generate the figures for this column, we looked at the total number of tokens (after stopword removal and stemming) in the clean and the full index across all the documents in a given collection. Table 1 clearly shows that tags, their attributes, scripts and similar noise account for a large proportion of all these Web collections. For ClueWeb09B, a massive 84.28% of all the terms in the collection are eliminated via the cleaning process. As expected, the number of distinct terms is also substantially smaller in the clean index than in the full index for all collections. While the impact of this cleaning on retrieval effectiveness will be studied in detail in Section 6, these numbers show that noise removal is a good idea simply from the perspective of computational resources required to process a collection.

4 APPROACHES INVESTIGATED

We now turn to the retrieval models that we considered in this study. We selected language model based retrieval methods, with two different smoothing techniques, namely Jelinek-Mercer smoothing, and Dirichlet smoothing [75–77]. These techniques are popular among researchers as baseline

[12]https://jsoup.org/

TREC collection	No. of docs	Markup-to-Length Ratio	No. of Distinct Terms	
			Full	Clean
WT10G	1,692,096	43.67%	10,055,958	7,282,492
GOV2	25,205,179	68.09%	85,590,568	59,513,645
ClueWeb09B-Spam 70	29,038,220	81.33%	151,574,283	43,487,205
ClueWeb09B	50,220,423	84.28%	418,246,153	118,442,851

Table 1: Statistics relating to the full and clean indexes for various Web collections.

retrieval methods [37, 54, 73, 74, 78]. To test the effect of document preprocessing on probabilistic and information theoretic methods, we also experimented with BM25 [60, 61] and Divergence from Randomness [2] retrieval models. Finally, we also studied the impact of noise removal on query expansion based on the relevance based language model [46, 48]. In the rest of this section, we briefly review these models and methods.

4.1 Language Modeling

The general idea of language modeling based retrieval can be described in the following way. Let Q be a query and d be a document. Let $\mathcal{D}$ represent the language model estimated from d. Then the score of document d with respect to query Q is given by $p(Q|\mathcal{D})$. The language model $\mathcal{D}$ associated with the document d is usually approximated by a unigram language model, i.e., $\mathcal{D} = \{p(w_i|d)\}_{i\in[1,|V|]}$ where V is the size of the vocabulary, and $p(w_i|d)$ represents the probability of randomly picking word w_i from document d. Thus, the retrieval score of d for a given query Q can be written as

$$\begin{aligned} \text{Score}(Q,d) &= p(Q|\mathcal{D}) \\ &= \prod_{q\in Q} p(q|d) \end{aligned} \tag{1}$$

4.1.1 Jelinek-Mercer Smoothing. To overcome the zero probability problem (when a query term is missing from d, its score becomes zero according to Equation 1), the language model $\mathcal{D}$ is smoothed by interpolating the Maximum Likelihood Estimate (MLE) of $p(w_i|d)$ with a background language model, estimated from the entire collection C, as in Equation 2.

$$\begin{aligned} p(Q|\mathcal{D}) &= \prod_{q\in Q} \left[(1-\lambda)p(q|d) + \lambda p(q|C)\right] \\ &= \prod_{q\in Q} (1-\lambda)\frac{tf(q,d)}{|d|} + \lambda\frac{cf(q)}{|C|} \end{aligned} \tag{2}$$

Here, $tf(q,d)$ and $cf(q)$ respectively indicates the count of term q in the document, and in the whole collection. $|d|$ and $|C|$ are the size of the document and the collection respectively. $\lambda = [0,1]$ is the interpolation parameter. This language modelling based retrieval model (Equation 2) is known as language model with Jelinek-Mercer smoothing or linear smoothing, and is referred to as JM or, LM-JM in the rest of this article.

4.1.2 Dirichlet Smoothing. Another smoothing method that is also used by the researchers is the Dirichlet prior smoothing method that uses Bayesian estimation instead of MLE. The mathematical form of this model is similar to the language model with Jelinek-Mercer smoothing (see Equation 2), except the interpolation parameter has a dynamic coefficient that changes according to document length (see Equation 3).

$$P(Q|\mathcal{D}) \quad = \quad \prod_{q \in Q} \frac{tf(q,d) + \mu p(q|C|)}{|d| + \mu} \tag{3}$$

Here, $tf(q,d)$ is the term frequency of q in document d, $|d|$ is the document length, and $p(q|C|)$ is the collection probability of q in the whole collection. μ is the interpolation parameter, and can be interpreted as the pseudo counts of words introduced through the prior. The practice is, μ is set in the range $[100, 5000]$. The language model based retrieval model presented in Equation 3 is known as language model with Dirichlet prior smoothing. In the rest of this article, Dir and LM-Dir are used interchangeably to refer to the same model that is presented in Equation 3.

4.2 BM25

BM25 is a traditional probabilistic retrieval model [60, 61] with better length normalization factors. Equation 4 mathematically presents the BM25 model to score a document d for a given query Q.

$$\text{Score}(Q,d) \quad = \quad \sum_{q \in Q} \log \frac{D - df(q) + 0.5}{df(q) + 0.5} \frac{tf(q,d)(k_1+1)}{tf(q,d) + k_1 \dot{(}1 - b + b\frac{|d|}{avgdl})} \tag{4}$$

In Equation 4 D is the number of documents in the collection, $df(q)$ is the number of documents in the collection containing the term q, $tf(q,d)$ corresponds to the count of term q in document d, and $avgdl$ is the average document length of the collection. b is a length normalization parameter, and k_1 is a positive tuning parameter that calibrates the document term frequency scaling [60].

4.3 Divergence from Randomness

The Divergence from Randomness (DFR) model has a number of associated parameters. For a given query $Q = \{q_1, \ldots q_k\}$, the within document term weight of a query term q ($q \in Q$) is defined as:

$$\text{w(q,d)} = \frac{cf(q) + 1}{df(q)(tfn + 1)} \big(tfn. \log_2 . \frac{D+1}{cf(q) + 0.5}\big) \tag{5}$$

where,

- $cf(q)$ - collection frequency of q in the whole collection
- $df(q)$ - document frequency of q in the collection
- D is the number of document in the collection
- tfn is the normalised term frequency, and defined as:

$$tfn = tf(q) \cdot \log_2(1 + c \cdot \frac{avgdl}{|d|})$$

  with c being a parameter whereas, $|d|$ and $avgdl$ denotes the document length and average document length respectively.

The primary focus of this work is to study the effect of metadata in different retrieval models. Hence we are not going into the non-trivial details of the different parameters and normalization factor of DFR models. For a comprehensive discussion of DFR model , please refer to [1, 2, 45].

4.4 Relevance Feedback

Similar to initial retrieval methods, retrieval involving query expansion (QE) may also suffer from performance variations due to inclusion or exclusion of metadata in the index. To study the variation in performance of QE, we choose relevance model based query expansion [46, 48], an association based method that remains an effective and widely used approach for query expansion.

In this section, we describe RLM, a popular acronym for relevance model. For a given query $Q = q_1, \dots q_k$, the relevance model hypothesises that there is a latent probability distribution R that generates both the query Q, and its relevant documents. The relevance model R is then approximated on the basis of terms of Q, and set of known relevant documents. The terms of R with large probability weights can then be selected as the potential expansion terms. The better the estimation of R, the better performance of the query expansion technique. In the absence of training data, M top ranked pseudo-relevant documents are considered as the set of relevant documents. The terms of query Q serves as the exclusive evidence about the relevance model, that is $q_1, \dots q_k$ are certainly generated from R. Thus, the probability density function of RLM, or the probability of sampling a term w from R, denoted by $P(w|R)$, is approximated by $P(w|Q)$, the conditional probability of observing w along with query terms $q_1, \dots, q_k$.

$$P(w|R) \approx P(w|Q) = \frac{P(w,Q)}{P(Q)} \sim P(w,Q) = P(w, q_1, \dots q_k) \tag{6}$$

Following Equation 6, estimating the probability density function $P(w|R)$ boils down to estimating the joint probability of observing w together with query terms $q_1, \dots, q_k$. The joint probability $P(w,Q)$ is then estimated using *independent and identically distributed* (IID) sampling.

In IID sampling, the term w is sampled conditionally, together with the query terms $q_1, \dots, q_k$ from the same distribution, underlying a top ranked document. Therefore the probability estimation of $P(w|R)$ will follow Equation 7, where D is the set of M (pseudo-)relevant documents.

$$P(w|R) \approx P(w,Q) = \sum_{d \in D} P(w|d) \prod_{q \in Q} P(q|d) \tag{7}$$

RLM [48] only considers the aspect of a term in the initial retrieved document list, without giving any special importance to the terms which are present in the query. The RM3 method [46] separately takes into account the original query terms, by linearly interpolating the relevance model with the query likelihood model (see Equation 8). RM3 was shown to perform significantly better than the traditional relevance model.

$$P'(w|R) = (1-\alpha)\Big(\sum_{d \in D} P(w|d) \prod_{q \in Q} P(q|d)\Big) + \alpha P(w|Q) \tag{8}$$

In Equation 8, $\alpha \in [0,1]$ is the linear interpolation parameter, and query likelihood model $P(w|Q)$ is computed using maximum likelihood estimation (MLE).

From the estimated probability distribution model R, top N terms with the highest probability distribution weights $P'(w|R)$ are chosen to be the expansion terms (with N being a parameter). The weight of the N expansion terms are then sum normalized to one before performing the retrieval with query expansion. Thus it is crucial for relevance model estimation to assign high weights to terms which are exclusive to the relevant documents, and to avoid assigning significant weights to common terms. Therefore, any form of filtering of the documents (in our case, metadata exclusion) may lead to deviation in performance.

For performing the initial retrieval, and retrieval with the expanded query, any retrieval model can be used. Following the trend in literature ([46, 48, 51]), as RLM is a language model based QE method, we perform both, baseline and expanded retrievals using the language model with the two different smoothing methods, JM and Dir.

In the following section, we describe the basic experimental setup that we followed for these experiments. Also, the dataset for the evaluation as well as, the evaluation metrics used are outlined. Later, in Section 6, the empirical evaluation of the different methods used for indexing is presented.

TREC collection	Topic Set (Title only)	RunID	Number of topics	Average topic length (in words)
WT10G	TREC 9 Web Track Topics 451-500	TREC 9	50	3.48
	TREC 10 Web Track Topics 501-550	TREC 10	50	4.88
GOV2	TREC 2004-2006 Terabyte Track Topics 701-850	GOV2	150	3.08
ClueWeb09B - Spam70	TREC 2009-2012 Web Track Topics 1-200	CW09B-S70	200	2.48
ClueWeb09B	TREC 2009-2012 Web Track Topics 1-200	CW09B	200	2.48

Table 2: Overview of datasets used in our experiments. RunID indicates the name used for specifying the experimental results on that topic set.

5 EXPERIMENTAL SETUP

We used Lucene 5.3.1[13] for our experiments. The document preprocessor described in Section 3 was incorporated into the Lucene indexing pipeline. We also implemented RM3 [46], the relevance model based query expansion method, within Lucene. The source code of our implementations is availablel[14]. For both the clean and the full index, common terms were eliminated using the SMART stopword list [62]; the remaining words were stemmed using Porter's stemmer. For retrieval, Lucene's native implementations of the baseline methods (Section 4) were used. Only the title field of queries was considered. Retrieval effectiveness was measured and compared using Mean Average Precision (MAP), but other evaluation metrics (e.g., *P@5* and *recall*) exhibit similar variations as MAP. All the significance tests are performed using paired t-test at 95% confidence level ($p < 0.05$).

5.1 Dataset Overview

Since the presence of meta-content (referred to as *noise*) is mostly a concern for Web corpora, we used TREC Web datasets — specifically WT10G, GOV2 and ClueWeb09B — for our experiments. WT10G is a well crafted Web collection containing about 1.5 million Web documents [8]. It was used for the TREC 9 and 10 Web Track tasks [22, 23]. GOV2 is a collection of 25 million webpages from the .gov domain. It was used in the TREC Terabyte Track [10, 14, 15]. ClueWeb09 - category B, containing about 50 million Web documents in English, was used for the Web Track at TREC from 2009 to 2012 [16–19]. As expected in a large enough sample of pages crawled from the Web, this collection contains a non-trivial proportion of spam documents. A spam filter for this collection was presented in [21][15]. For our experiments with baseline retrieval methods, we used both the original collection, and a spam-filtered version. The filtered version was created by eliminating all documents with a spam score of less than 70% (as suggested in https://plg.uwaterloo.ca/~gvcormac/clueweb09spam/ and used in [28, 54, 71, 78]). The results of experiments on full ClueWeb09 - category B and its spam filtered (with 70%) version are reported in Table 5 with the short-hand notation CW09B and CW09B-S70 respectively. However, we noticed

[13] http://lucene.apache.org/
[14] https://github.com/dwaipayanroy/WebDoc_Indexer_Retriever_QE
[15] https://plg.uwaterloo.ca/~gvcormac/clueweb09spam/

Retrieval Model	Range of Parameter Variation
LM-JM	$\lambda = \{0.1 - 0.9\}$
LM-Dir	$\mu = \{100, 200, 500, 1000, 1500, 2000, 5000\}$
BM25	$k_1 = \{0.5 - 1.5\}$ $b = \{0.1 - 0.9\}$
DFR	BasicModel = *{BE, D, G, IF, In. Ine, P}* Info. gain normalization = *{L, B, NoNorm}* Length normalization = *{Uni., InvLen., Dir., Zip., NoNorm}*

Table 3: Different parameter/component settings of various retrieval methods tried during cross validation.

that a number of judged relevant documents were eliminated by this process. Further, the results on both the filtered collection and the full collection follow a similar pattern by and large. Therefore, for the experiments on Relevance Feedback (presented in Sections 6.2), we report only the results for the entire ClueWeb09 - Category B collection without any spam-filtering. An overview of these datasets is provided in Table 2.

5.2 Parameter Tuning

The retrieval models used in this study have a number of associated parameters. As discussed in Sections 4.1 and 4.2, the language modeling based methods (LM-JM and LM-Dir) have one parameter each (λ and μ respectively), while BM25 has two parameters (k_1 and b). The DFR model is made up of three components, namely, *basic-model*, *information-gain normalization*, and *length normalization* ([1, 2] contains a detailed discussion about these components). All parameters of these baseline models were tuned separately for each topic set listed in Table 2 using 5-fold cross validation. Further, all the parameters were separately tuned for the clean and full indexes. The linear smoothing parameter λ of LM-JM was varied from 0.1 to 0.9 in steps of 0.1; the Dirichlet parameter μ was varied over the range $\{100, 200, 500, 1000, 2000, 2500, 3000, 5000\}$. The BM25 parameters ($k_1$ and b) were varied over $[0.1, 2.0]$ and $[0.0, 1.0]$ respectively, in steps of 0.1. For DFR, all possible combinations of the three different components were explored. The range of parameters tried for each baseline retrieval model is presented in Table 3.

For experiments involving query expansion (Section 4.4), the RM3 parameters were also selected using 5-fold cross validation. These parameters were tuned separately for the two document smoothing methods (LM-JM and LM-Dir), since these two methods are known to behave differently for queries of different lengths [76]. We varied the number of expansion terms in the range from 50 to 90, and from 30 to 70 separately for LM-JM and LM-Dir respectively. We fixed the number of feedback documents at 10. The query mixing parameter (α of Equation 8) was varied in the range $\{0.3 - 0.7\}$ in steps of 0.1 for all the QE experiments.

The baseline retrieval results show that the performance of LM-JM drastically deteriorates for λ values greater than 0.3. Similarly, increasing μ beyond 2000 resulted in gradual degradation of MAP for LM-Dir. Therefore, to keep the overall number of parameter combinations to be tried for RM3 to a tractable number, we limited the range for λ and μ to $\{0.1, 0.2, 0.3\}$ and $\{1000, 1500, 2000\}$ respectively. For a particular retrieval model (LM-JM or LM-Dir), the smoothing parameter selected for baseline retrieval was also used when performing retrieval with the expanded query. Table 4 shows the range of parameter values that were tried during cross validation for RM3.

Model	Smoothing Parameter Range	RM3 Parameters		
		M - # fdbk docs.	N - # expans. terms	α - query wt.
LM-JM	$\lambda = \{0.1 - 0.3\}$	10	50, 60, 70, 80, 90	$\{0.3 - 0.7\}$
LM-Dir	$\mu = \{1000, 1500, 2000\}$	10	30, 40, 50, 60, 70	$\{0.3 - 0.7\}$

Table 4: Parameter settings tried during cross-validation for LM-JM and LM-Dir based RM3 feedback model.

Collection	Content Used	LM-JM	LM-Dir	BM25	DFR
TREC 9	Full	0.1819	0.2127	0.2173	0.2039
	Clean	0.1563	0.2237	0.2332	0.2162
	% change	**-14.07**	+5.17	**+7.32**	**+6.03**
TREC 10	Full	0.1643	0.2047	0.1953	0.1825
	Clean	0.1445	0.2072	0.1964	0.1936
	% change	**-12.05**	+1.22	+0.56	**+6.08**
GOV2	Full	0.2298	0.2705	0.2797	0.2755
	Clean	0.2181	0.2953	0.2996	0.2897
	% change	-5.09	**+9.17**	**+7.11**	**+5.15**
CW09B-S70	Full	0.0907	0.1184	0.1269	0.1377
	Clean	0.0904	0.1558	0.1574	0.1646
	% change	-0.33	**+31.59**	**+24.03**	**+19.54**
CW09B	Full	0.1055	0.1266	0.1476	0.1560
	Clean	0.1122	0.1738	0.1941	0.1962
	% change	**+6.35**	**+37.28**	**+31.50**	**+25.77**

Table 5: Comparison of Mean Average Precision (MAP) for different retrieval models and indexing schemes on different topic sets. % changes in bold denote changes that are statistically significant (based on a paired t-test at the 5% level).

6 RESULTS AND DISCUSSIONS

We first present the results obtained using the full and clean indexes for various baseline retrieval approaches. Next, in Section 6.2, we look at the effect of noise cleaning on RM3-based query expansion.

6.1 Baseline Retrieval

Table 5 presents the performance of four different retrieval strategies when the full and clean indexes are used. Our observations are summarised below.

- On all collections except ClueWeb09B, noise removal has a (sometimes strong) detrimental effect on LM-JM. This was, prima facie, an unexpected result, but one that is explained below. Noise removal has no effect if spam is filtered out from the ClueWeb09B collection. Cleaning improves the performance of LM-JM only for the full ClueWeb09B collection, but the margin of improvement is much lower than what is observed for the other retrieval models.
- For LM-Dir and BM25, noise removal generally does not seem to have much impact when the test collection is relatively small and noise-free. Specifically, for the TREC 10 topic set, cleaning has minimal effect on the performance of LM-Dir and BM25. As the collections

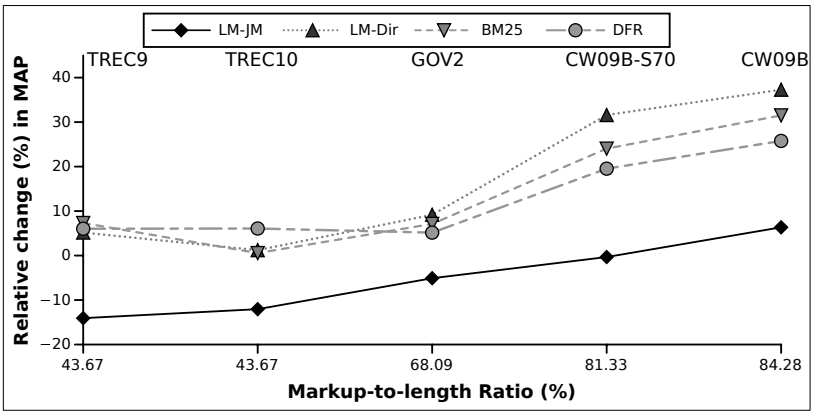

Fig. 3: Relative change in MAP versus percentage of noise in the collection.

grow in size and become progressively noisier, cleaning has a more pronounced positive effect. For ClueWeb09B, cleaning improves MAP by as much as 30% or more.

- The effect of cleaning on DFR is significant for all the topic sets. As expected, the maximum difference is observed for the complete, unfiltered ClueWeb09B collection.
- Combining the information in Table 5 with the third column of Table 1, we note that all four retrieval models exhibit a common trend: cleaning plays an increasingly useful role as collections become noisier. This trend is shown graphically in Figure 3: the X-axis (not to scale) shows the percentage of markup content in different document collections (see the third column of Table 1), and the Y-axis indicates relative change in MAP for the respective collection. Note that TREC 9 and 10 have the same markup-to-length ratio, as both of them use the same WT10G test collection.
- Spam filtering removes the most noisy documents, thereby taking care of some of the effects of noise cleaning. Cleaning thus produces somewhat smaller improvements for LM-Dir, BM25 and DFR on CW09B-S70 compared to the unfiltered ClueWeb09B collection. Nevertheless, the improvements are substantial in absolute terms (around 20% or more) and statistically significant.
- In case of language model based retrieval methods, Dirichlet smoothing consistently outperforms Jelinek-Mercer smoothing. This observation is consistent with the findings of Zhai and Lafferty [76, 77].
- When performance across all collections is taken into account, BM25 seems to be the most consistently effective retrieval model.

We now turn our attention to the apparently counter-intuitive behaviour of LM-JM. For the WT10G collection (with topic sets TREC 9 and TREC 10), using the noisy, full index consistently produces *significantly* better results if JM smoothing is used (for the GOV2 collection, the noisy index also yields better results, but the improvements are not statistically significant). This is best explained by considering the difference between JM and Dirichlet. Dirichlet smoothing (Equation 3) is document-length dependent: maximum likelihood estimates (MLEs) are less reliable for short documents, and are more heavily smoothed; conversely, MLEs from longer documents are more reliable, and are smoothed to a smaller extent. In contrast, JM smoothing is document length independent.

Consider a single-term query $Q = \{q\}$ and two documents d_1 and d_2. Suppose that d_1 is very short (too short to be relevant), but contains a chance occurrence of the query term q; also suppose that d_2 is a longer, useful document that contains more occurrences of q, but also many more additional terms. The Maximum Likelihood Estimate (MLE) of $p(q|d)$ would typically be much higher for d_1 than for d_2. When JM smoothing is used (Equation 2), d_1 would get a much higher score compared to d_2. Thus, JM has a bias in retrieval towards short documents. It is well-known (from studies by Singhal et al. [63, 64], for example) that retrieval functions (such as JM) that are biased in favour of short documents perform relatively poorly. This explanation is also consistent with earlier analyses explaining the advantage of Dirichlet over JM [50, 65].

This explanation can be extended to account for why JM performs better over a full index. Consider TREC topic 503 (*Vikings in Scotland?*), and two documents d = WTX103-B39-174 (shown in Figure 1) and d' = WTX075-B17-106 (shown in Figure 2). d is short, uninformative and non-relevant, while d' contains less metadata, and is relevant. When markup / meta-content is removed, short documents like d become very short (with possibly as few as 3-5 words). If such an extremely short document contains even a single occurrence of one query term q, $P_{MLE}(q|d)$, the maximum likelihood estimate of the probability of q being generated by the document model d, becomes unnaturally high. For our example, $P_{MLE}(scotland|d) = 0.25$. In contrast, $P_{MLE}(scotland|d')$ has a far more modest value of 0.026, even though d' is actually useful, and has as many as 34 occurrences of *scotland*. From Equation 2, it is clear that d would be ranked far above d' if JM smoothing were used.

When full documents are used, all documents appear longer because each document contains its fair share of noise. Thus, the values of $P_{MLE}(q|d)$ are more moderate, and documents with little useful content no longer have an unreasonable advantage. Therefore, retrieval effectiveness improves.

Given this anomalous behaviour of LM-JM, and given that researchers continue to use LM-JM, the recommendations of Zhai and Lafferty [76, 77] notwithstanding (e.g. see [6, 30, 35, 36, 55, 58]), our experiments thus demonstrate the importance of document parsing from the reproducibility perspective used for a particular study.

6.2 Pseudo Feedback based Query Expansion

In this section, we examine the differences in improvement obtained when either the full or the clean index is used during pseudo relevance feedback (PRF) based QE. Specifically, we consider RM3 based QE (discussed in Section 4.4) for our study. In principle, there are 3 indexes involved during PRF-based QE: the index used for initial retrieval (I_1), the index from which expansion terms are selected (I_2), and the index used for retrieval after expansion (I_3). In order to reap the full benefits of QE, I_2 and I_3 should, of course, be the same. We use the term *Baseline Index* to refer to I_1 and *Feedback Index* to refer to I_2 and I_3. Since the clean or the full index can be used either as the Baseline Index and / or the Feedback Index, we get a total of four possible combinations. We refer to these combinations as full-full, full-clean, clean-full, and clean-clean. The first part of a combination's name signifies which index was used for initial retrieval, while the second specifies which index was used during QE and the final retrieval. For example, "full-clean" denotes a configuration in which the initial retrieval was performed from the *Full-index*, while the *Clean-index* was used during selection of expansion terms, and during the final retrieval with the expanded query ($I_1 \leftarrow$ Full-index, $I_2, I_3 \leftarrow$ Clean-index).

It is a known fact that the relevance model is prone to choosing common terms [20, 24, 44]. Dropping stopwords from the collection during indexing partially addresses the problem. However, terms which are common in the top-ranked documents, but not present in the stopword list (e.g., HTML tags in web documents) are quite often selected as expansion terms by the relevance model

due to their high association with query terms in these documents. For this reason, noise cleaning is likely to be important for QE: it is expected to eliminate at least some inappropriate terms from being candidate expansion terms.[16]

The results for RM3 (Table 6) are conform to our expectations. For a given baseline index (either clean or full), using the clean index for feedback works better than using the full index. The improvement is sometimes minuscule (e.g., full-clean vs. full-full for LM-Dir on GOV2), but often significant. As discussed above, this is easily explained: tags and other noisy terms are included in the query if the full index is used during QE. In contrast, the terms selected for QE from the clean index are generally much more related to the actual intent of the query.

Table 7 illustrates this phenomenon using one example query from each of the four test collections used in our experiments. For these examples, LM-JM was used for initial retrieval from the full index. For each query, the top 15 expansion terms (having the highest $P(w|r)$ weights according to Equation 8) are shown. Note that, for a particular query, these expansion terms were taken from the *same* set of pseudo-relevant documents. Expansion terms were selected from the full index for the full-full configuration, while the clean index for the same documents was used to construct the full-clean query. It is clear from the table that many / most of the terms selected are tags, when the full index is used for expansion.

For both topic sets that use the WT10G collection (TREC 9 and 10), the best results for LM-JM are obtained when the full index is used for the initial retrieval (as this provides a much better starting point than the clean index), and the clean index is used for feedback. Further, these results are statistically significantly better than the RM3 results for the other combinations (full-full, clean-full and clean-clean). In case of GOV2, although the initial retrieval was inferior when the clean index was used (a 5% difference in MAP that is not statistically significant), the clean-clean combination produced the best performance among all combinations. For ClueWeb09B, initial retrieval from the clean index produces significantly better results; not surprisingly, the use of the clean index for both the initial retrieval as well as feedback yields the best results.

When the baseline retrieval is performed using LM-Dir, the clean index produces better results for all topic sets (See Figure 3). Once again, it is not surprising that post-QE performance is generally best for the clean-clean combination. The only exception to this pattern is that, the full-clean combination achieved the best performance for TREC 10, but it is only marginally better than the clean-clean combination.

As a general observation from Table 6, we conclude that the removal of noise during indexing can produce significantly different results for QE-based retrieval methods, irrespective of the index chosen for initial retrieval. Thus, the choice of parsing method once again turns out to be a significant detail that should be mentioned to ensure the reproducibility of experiments involving QE methods.

6.2.1 When does cleaning hurt performance? As with many other "standard" IR techniques (such as stemming or stopword removal), noise removal is not beneficial for all queries. A per-query analysis of the performance variation of RM3 across document preprocessing strategies (specifically, full-full and full-clean) is shown graphically in Figure 4. In the figure, each vertical bar corresponds to one query, and represents the difference in AP observed when the query is expanded using the

[16]One might argue that these 'useless' expansion terms do not hurt effectiveness because of the IDF factor that comes into play during the final retrieval using the expanded query. However, if the number of expansion terms is kept fixed, then these terms would elbow out more useful terms from the expanded query. In order to reap the potential benefits of QE in the presence of such terms, one would need to increase the total number of expansion terms. This would adversely affect efficiency, if not effectiveness.

Collection	Baseline index	Feedback index	JM			Dir		
			Baseline	RM3	% change	Baseline	RM3	% change
TREC 9	Full	Full	0.1819	0.1751	-3.74	0.2127	0.2226	+4.65
		Clean		**0.2115**^{1,3,4}	+16.27		0.2352	+10.58
	Clean	Full	0.1563	0.1628	+4.16	0.2237	0.2360	+5.50
		Clean		0.1866	+19.39		**0.2404**^{1}	+7.47
TREC 10	Full	Full	0.1643	0.1879	+14.36	0.2047	0.2227	+8.79
		Clean		**0.2161**^{1,3,4}	+31.53		**0.2353**^{1}	+14.95
	Clean	Full	0.1445	0.1673	+15.78	0.2072	0.2285	+10.28
		Clean		0.1861	+28.79		0.2320	+11.97
GOV2	Full	Full	0.2298	0.2459	+7.00	0.2705	0.2920	+7.95
		Clean		0.2563	+11.53		0.2943	+8.80
	Clean	Full	0.2181	0.2266	+3.90	0.2953	0.3156	+6.87
		Clean		**0.2638**^{1,2,3}	+20.95		**0.3269**^{1,2,3}	+10.70
CW09B	Full	Full	0.1055	0.1236	+17.16	0.1266	0.1277	+0.87
		Clean		0.1360	+28.91		0.1422	+12.32
	Clean	Full	0.1122	0.1249	+11.32	0.1738	0.1759	+1.20
		Clean		**0.1540**^{1,2,3}	+37.25		**0.1882**^{1,2,3}	+8.29

Table 6: Comparison of Mean Average Precision (MAP) for baseline and RM3 for different smoothing methods and indexing schemes on different topic sets. The RM3 results in bold denote the maximum improvement over baseline as compared to the other configurations for the same topic set and retrieval model. Note that for all collections, the maximum improvements are achieved when the clean index is used for pseudo relevance feedback. The superscripts ^{1,2,3,4} denote a statistically significant difference between the corresponding value and the MAP for the full-full, full-clean, clean-full and clean-clean combinations respectively.

clean and the full index. Bars above the X-axis correspond to queries for which expansion from the clean index works better.

From Figure 4, we note that for some 'unusual' queries, expansion from the noisy index results in better performance. It is possible that the meta-information eliminated by our cleaning method actually contains useful expansion terms for such queries. This is not surprising, as the intended use of the meta tags is to provide additional information about a webpage that increases its chances of matching a query about its content (for example, users may include their aliases in the form of meta-content on their personal home page). In our experiments, meta-content was either treated at par with actual content, markup and scripts (for the full index), or entirely filtered out (for the clean index). This cleaning method was possibly too aggressive in some cases.

Table 8 lists a few examples of these 'unusual' queries. A closer look at these queries reveals a counter-intuitive pattern, however. For all three examples, the highly-weighted expansion terms in the full-full queries correspond to tags and tag-attributes, almost without exception (see Table 9). This is similar to the pattern observed in Table 7. In spite of this, the full-full queries yield better Average Precision than the full-clean queries. Query 67 from the ClueWeb09B dataset is particularly baffling: to a human, many of the expansion terms in the full-clean query appear to be strongly related to the query topic (*cholesterol*, *level*, *lipoprotein*, *high*). The performance variation for query 484 is somewhat easier to explain. Since most of the expansion terms in the full-full query are tags, these terms do not add any additional discriminatory power to the query. As a result, the original, unexpanded query and the full-full query yield roughly comparable results. The

499 - "pool cue"			
full index		clean index	
w	$P(w\|R)$	w	$P(w\|R)$
pool	0.3055	pool	0.3215
cue	0.3032	cue	0.3127
td	0.0369	tabl	0.0111
br	0.0297	1996	0.0108
href	0.0295	ball	0.0100
font	0.0165	page	0.0091
tr	0.0126	time	0.0078
center	0.0126	inform	0.0077
http	0.0105	web	0.0071
img	0.0102	home	0.0067
li	0.0101	plai	0.0066
src	0.0100	game	0.0063
align	0.0097	make	0.0062
html	0.0092	line	0.0061
width	0.0092	year	0.0061

529 - "history on cambodia?"			
full index		clean index	
w	$P(w\|R)$	w	$P(w\|R)$
cambodia	0.2151	cambodia	0.2455
histori	0.2113	histori	0.2301
td	0.0480	1	0.0310
href	0.0386	yahoo	0.0187
br	0.0374	countri	0.0174
font	0.0217	inform	0.0172
html	0.0211	lao	0.0168
1	0.0185	societi	0.0150
width	0.0167	cultur	0.0148
titl	0.0167	2	0.0129
tr	0.0164	search	0.0118
h3	0.0155	region	0.0102
center	0.0150	khmer	0.0098
http	0.0148	1996	0.0097
li	0.0138	centuri	0.0096

146 - "sherwood regional library"			
full index		clean index	
w	$P(w\|R)$	w	$P(w\|R)$
librari	0.1391	librari	0.1976
region	0.1333	region	0.1532
sherwood	0.1333	sherwood	0.1500
href	0.0435	link	0.0183
td	0.0406	1	0.0173
class	0.0368	home	0.0156
div	0.0312	alexandria	0.0151
li	0.0300	2008	0.0144
http	0.0278	counti	0.0138
tr	0.0216	inform	0.0133
span	0.0206	number	0.0131
titl	0.0196	locat	0.0129
0	0.0181	2	0.0127
width	0.0164	contact	0.0125
id	0.0143	virginia	0.0124

847 - "portug world war ii"			
full index		clean index	
w	$P(w\|R)$	w	$P(w\|R)$
ii	0.0999	war	0.1065
portug	0.0999	ii	0.0999
war	0.0999	portug	0.0999
world	0.0999	world	0.0999
td	0.0860	1	0.0368
tr	0.0316	state	0.0236
font	0.0314	data	0.0225
href	0.0306	2	0.0218
0	0.0258	inform	0.0161
width	0.0216	3	0.0161
align	0.0198	servic	0.0150
img	0.0177	4	0.0141
1	0.0166	5	0.0128
br	0.0165	2003	0.0126
class	0.0163	nation	0.0121

Table 7: Top 15 terms of some expanded queries, with associated weights. LM-JM was used for the initial retrieval from the full index (except for CW09B, for which the clean index was used for initial retrieval). Queries were expanded using RM3. Notice that, most of the expansion terms, when selected from the full index, are unimportant terms from meta-content.

terms added to the full-clean query are meaningful, but cause the query to drift. Consequently, the AP obtained using this query is considerably lower than the baseline. We hope to study these queries more systematically in future work in order to quantify the above hypothesis regarding useful meta-content, as well as our observations regarding query drift.

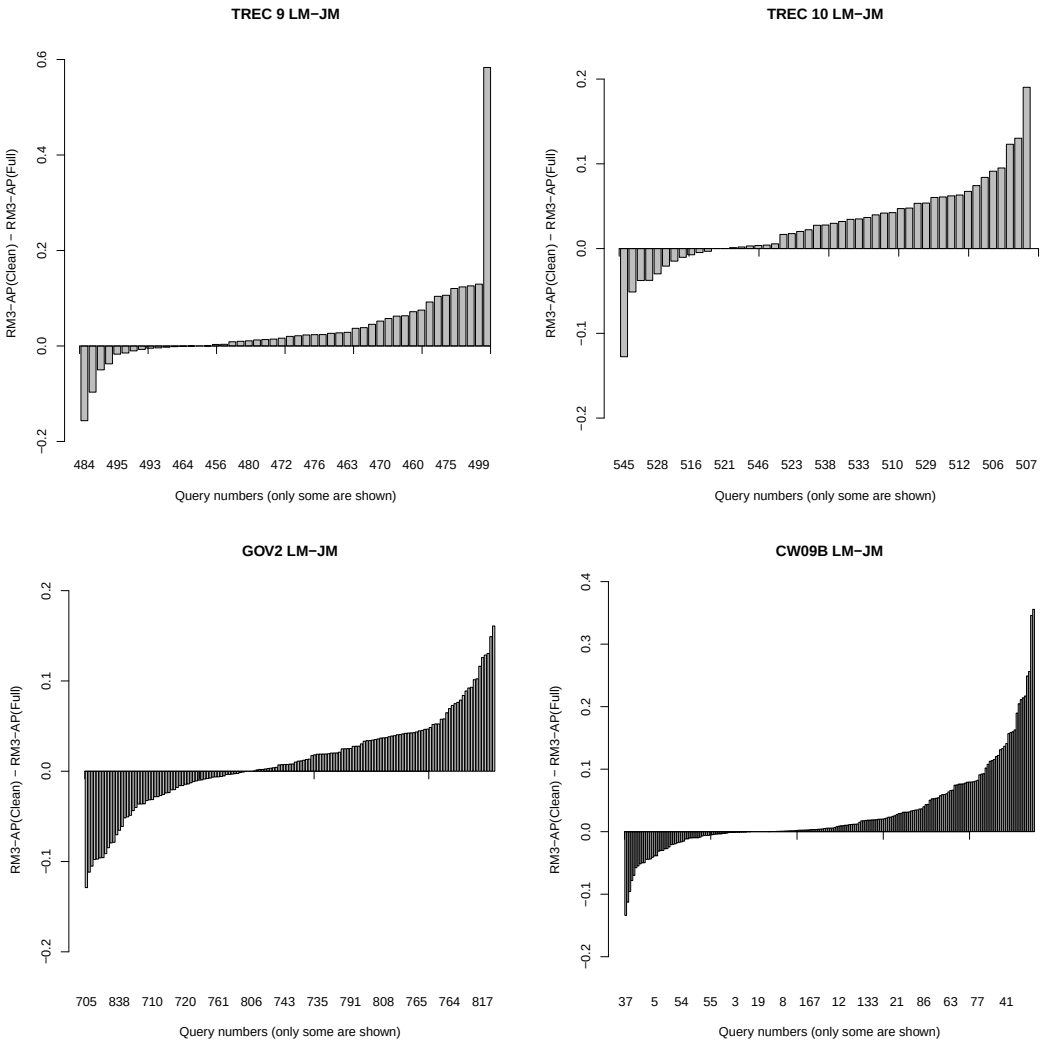

Fig. 4: Difference in AP for expanded queries from different topic sets. LM-JM was used for the initial retrieval from the full index (except for CW09B, for which the clean index was used for initial retrieval). Queries were expanded using the full and the clean indexes, and the difference in performance is plotted along the Y-axis.

Query no.	Query text (processed title)	Collection	QE-AP (full-full)	QE-AP (full-clean)
67	vldl levels	CW09B	0.4633	0.3675
484	auto skoda	TREC 9	0.2693	0.1792
705	iraq foreign debt reduction	GOV2	0.2712	0.1423

Table 8: Examples of queries for which expansion using the full index gives better results.

67 - "vldl level"	
full index	clean index
href	cholesterol
td	1
class	2008
li	2
div	level
http	home
tr	page
titl	3
span	articl
0	site
width	lipoprotein
id	high
font	inform
style	link
br	free

484 - "auto skoda"	
full index	clean index
td	1
br	page
href	skoda
font	2
center	site
tr	0
http	provid
li	1996
1	3
img	real
src	download
align	space
0	inform
width	servic
size	movi

705 - "iraq foreign debt reduct"	
full index	clean index
td	0
tr	1
href	state
font	2
0	2002
width	inform
align	3
imag	2003
img	17
1	4
br	u.
src	servic
alt	5
border	program
height	nation

Table 9: Top 15 terms of some expanded queries. LM-JM was used for the initial retrieval from the full index (except for CW09B, for which the clean index was used for initial retrieval). Queries were expanded using RM3. Even though most of the expansion terms selected from the full index are HTML tags, these queries perform better than the expanded queries obtained using the clean index.

7 CONCLUSION

The main questions that our study was intended to address were introduced in Section 1. They are:

Question 1. Should documents be cleaned prior to indexing? How much difference does noise-removal make?

Question 2. What is the effect of noise removal on query expansion (QE) techniques?

We specifically focus on the TREC Web collections used for text retrieval research, since these contain a non-trivial amount of noise in the form of meta-tags, hyperlinks etc. How this noise is handled during indexing turns out to be an important issue from the point of view of reproducibility, as noise cleaning can significantly affect retrieval quality for some retrieval methods. Based on our experimental results, we conclude with the following observations in response to the above questions.

(1) For most retrieval models, documents should indeed be cleaned prior to indexing. Cleaning generally results in improvements in MAP. Depending on the retrieval model and the amount of noise present in a corpus, these improvements may be very substantial (we observed an improvement of about 37% in case of LM-Dir on the ClueWeb09B collection).
(2) Language modeling with Jelinek-Mercer smoothing exhibits anomalous behaviour with regard to the above observation. Other than on extremely noisy collections (ClueWeb09B), cleaning adversely affects MAP, sometimes significantly so. This can be explained using the fact that Jelinek-Mercer smoothing does not take document length into account.
(3) Using a clean index during pseudo relevance feedback based query expansion is also seen to be beneficial overall.

We observe from Figure 4, however, that our noise-cleaning method can be detrimental for some queries. For these queries, expansion using the original, "noisy" documents produces better results.

Some anecdotal analysis of this observation has been provided in Section 6.2. In future work, we would like to do a more thorough analysis of when and why this happens. On a related note, we would also like to explore methods that make use of the meta-content more carefully for Web corpora. Finally, the performance of fielded retrieval models on Web document collections may be worth investigating.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable comments which have helped to improve and clarify this work.

REFERENCES

[1] Giambattista Amati. 2003. *Probability models for information retrieval based on divergence from randomness.* Ph.D. Dissertation. University of Glasgow.

[2] Gianni Amati and Cornelis Joost Van Rijsbergen. 2002. Probabilistic Models of Information Retrieval Based on Measuring the Divergence from Randomness. *ACM Trans. Inf. Syst.* 20, 4 (Oct. 2002), 357–389. https://doi.org/10.1145/582415.582416

[3] Yael Anava, Anna Shtok, Oren Kurland, and Ella Rabinovich. 2016. A Probabilistic Fusion Framework. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. ACM, New York, NY, USA, 1463–1472. https://doi.org/10.1145/2983323.2983739

[4] Jaime Arguello, Matt Crane, Fernando Diaz, Jimmy Lin, and Andrew Trotman. 2016. Report on the SIGIR 2015 workshop on reproducibility, inexplicability, and generalizability of results (RIGOR). *ACM SIGIR Forum* 49, 2 (2016), 107–116.

[5] Timothy G. Armstrong, Alistair Moffat, William Webber, and Justin Zobel. 2009. Improvements that don't add up: ad-hoc retrieval results since 1998. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, Hong Kong, China, November 2-6, 2009*, David Wai-Lok Cheung, Il-Yeol Song, Wesley W. Chu, Xiaohua Hu, and Jimmy J. Lin (Eds.). ACM, 601–610. https://doi.org/10.1145/1645953.1646031

[6] Hosein Azarbonyad, Mostafa Dehghani, Maarten Marx, and Jaap Kamps. 2015. Time-Aware Authorship Attribution for Short Text Streams. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. ACM, New York, NY, USA, 727–730. https://doi.org/10.1145/2766462.2767799

[7] Leif Azzopardi, Yashar Moshfeghi, Martin Halvey, Rami S. Alkhawaldeh, Krisztian Balog, Emanuele Di Buccio, Diego Ceccarelli, Juan M. Fernández-Luna, Charlie Hull, Jake Mannix, and Sauparna Palchowdhury. 2017. Lucene4IR: Developing Information Retrieval Evaluation Resources Using Lucene. *SIGIR Forum* 50, 2 (Feb. 2017), 58–75. https://doi.org/10.1145/3053408.3053421

[8] Peter Bailey, Nick Craswell, and David Hawking. 2003. Engineering a Multi-purpose Test Collection for Web Retrieval Experiments. *Inf. Process. Manage.* 39, 6 (Nov. 2003), 853–871. https://doi.org/10.1016/S0306-4573(02)00084-5

[9] Saeid Balaneshin-kordan and Alexander Kotov. 2016. Sequential Query Expansion Using Concept Graph. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. ACM, New York, NY, USA, 155–164. https://doi.org/10.1145/2983323.2983857

[10] Stefan Büttcher, Charles LA Clarke, and Ian Soboroff. [n. d.]. The TREC 2006 Terabyte Track. https://trec.nist.gov/pubs/trec15/papers/TERA06.OVERVIEW.pdf

[11] Deng Cai, Shipeng Yu, Ji-Rong Wen, and Wei-Ying Ma. 2003. Extracting Content Structure for Web Pages Based on Visual Representation. In *Proceedings of the 5th Asia-Pacific Web Conference on Web Technologies and Applications (APWeb'03)*. Springer-Verlag, Berlin, Heidelberg, 406–417. http://dl.acm.org/citation.cfm?id=1766091.1766143

[12] Carlos Castillo and Brian D. Davison. 2011. Adversarial Web Search. *Found. Trends Inf. Retr.* 4, 5 (May 2011), 377–486. https://doi.org/10.1561/1500000021

[13] Carlos Castillo, Debora Donato, Luca Becchetti, Paolo Boldi, Stefano Leonardi, Massimo Santini, and Sebastiano Vigna. 2006. A Reference Collection for Web Spam. *SIGIR Forum* 40, 2 (Dec. 2006), 11–24. https://doi.org/10.1145/1189702.1189703

[14] Charles LA Clarke, Nick Craswell, and Ian Soboroff. [n. d.]. Overview of the TREC 2004 Terabyte Track. https://trec.nist.gov/pubs/trec13/papers/TERA.OVERVIEW.ps

[15] Charles LA Clarke, Falk Scholer, and Ian Soboroff. [n. d.]. The TREC 2005 Terabyte Track. https://trec.nist.gov/pubs/trec14/papers/TERABYTE.OVERVIEW.pdf

[16] Charles L. A. Clarke, Nick Craswell, and Ian Soboroff. 2009. Overview of the TREC 2009 Web Track. In *Proceedings of The Eighteenth Text REtrieval Conference, TREC 2009, Gaithersburg, Maryland, USA, November 17-20, 2009*. http://trec.nist.gov/pubs/trec18/papers/WEB09.OVERVIEW.pdf

[17] Charles L. A. Clarke, Nick Craswell, Ian Soboroff, and Ellen M. Voorhees. 2011. Overview of the TREC 2011 Web Track. In *Proceedings of The Twentieth Text REtrieval Conference, TREC 2011, Gaithersburg, Maryland, USA, November 15-18, 2011*. http://trec.nist.gov/pubs/trec20/papers/WEB.OVERVIEW.pdf

[18] Charles L. A. Clarke, Nick Craswell, and Ellen M. Voorhees. 2012. Overview of the TREC 2012 Web Track. In *Proceedings of The Twenty-First Text REtrieval Conference, TREC 2012, Gaithersburg, Maryland, USA, November 6-9, 2012*. http://trec.nist.gov/pubs/trec21/papers/WEB12.overview.pdf

[19] Charles L. A. Clarke, Ian Soboroff Nick Craswell, and Gordon V. Cormack. 2010. Overview of the TREC 2010 Web Track. In *Proceedings of The Eighteenth Text REtrieval Conference, TREC 2010, Gaithersburg, Maryland, USA, 2010*. http://trec.nist.gov/pubs/trec19/papers/WEB.OVERVIEW.pdf

[20] Stéphane Clinchant and Eric Gaussier. 2013. A Theoretical Analysis of Pseudo-Relevance Feedback Models. In *Proceedings of the 2013 Conference on the Theory of Information Retrieval (ICTIR '13)*. ACM, New York, NY, USA, Article 6, 8 pages. https://doi.org/10.1145/2499178.2499179

[21] Gordon V. Cormack, Mark D. Smucker, and Charles L. A. Clarke. 2011. Efficient and effective spam filtering and re-ranking for large web datasets. *Inf. Retr.* 14, 5 (2011), 441–465. https://doi.org/10.1007/s10791-011-9162-z

[22] Nick Craswell and David Hawking. 2004. Overview of the TREC 2004 Web Track. In *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004*. http://trec.nist.gov/pubs/trec13/papers/WEB.OVERVIEW.pdf

[23] Nick Craswell, David Hawking, Ross Wilkinson, and Mingfang Wu. 2003. Overview of the TREC 2003 Web Track. In *Proceedings of The Twelfth Text REtrieval Conference, TREC 2003, Gaithersburg, Maryland, USA, November 18-21, 2003*. 78–92. http://trec.nist.gov/pubs/trec12/papers/WEB.OVERVIEW.pdf

[24] Ronan Cummins. 2017. Improved Query-Topic Models Using Pseudo-Relevant Pólya Document Models. In *Proceedings of the 3rd ACM International Conference on the Theory of Information Retrieval, ICTIR 2017) Amsterdam, Netherlands, October 1-4, 2017*. http://dcs.gla.ac.uk/~ronanc/papers/cumminsICTIR17.pdf To appear.

[25] Ronan Cummins, Jiaul H. Paik, and Yuanhua Lv. 2015. A Pólya Urn Document Language Model for Improved Information Retrieval. *ACM Trans. Inf. Syst.* 33, 4 (2015), 21:1–21:34. https://doi.org/10.1145/2746231

[26] Zhuyun Dai, Chenyan Xiong, and Jamie Callan. 2016. Query-Biased Partitioning for Selective Search. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. ACM, New York, NY, USA, 1119–1128. https://doi.org/10.1145/2983323.2983706

[27] Mostafa Dehghani, Samira Abnar, and Jaap Kamps. 2016. The Healing Power of Poison: Helpful Non-relevant Documents in Feedback. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. ACM, New York, NY, USA, 2065–2068. https://doi.org/10.1145/2983323.2983910

[28] Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W. Bruce Croft. 2017. Neural Ranking Models with Weak Supervision. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, New York, NY, USA, 65–74. https://doi.org/10.1145/3077136.3080832

[29] Emanuele Di Buccio, Giorgio Maria Di Nunzio, Nicola Ferro, DK Harman, Maria Maistro, and Gianmaria Silvello. 2015. Unfolding off-the-shelf IR systems for reproducibility. In *Proc. SIGIR Workshop on Reproducibility, Inexplicability, and Generalizability of Results (RIGOR 2015)*.

[30] Faezeh Ensan and Ebrahim Bagheri. 2017. Document Retrieval Model Through Semantic Linking. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM '17)*. ACM, New York, NY, USA, 181–190. https://doi.org/10.1145/3018661.3018692

[31] Nicola Ferro. 2017. Reproducibility Challenges in Information Retrieval Evaluation. *J. Data and Information Quality* 8, 2 (2017), 8:1–8:4. https://doi.org/10.1145/3020206

[32] N. Ferro, F. Crestani, M.F. Moens, J. Mothe, F. Silvestri, G.M. Di Nunzio, C. Hauff, and G. Silvello (Eds.). 2016. *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016*. LNCS, Vol. 9626. Springer.

[33] Nicola Ferro, Norbert Fuhr, Kalervo Järvelin, Noriko Kando, Matthias Lippold, and Justin Zobel. 2016. Increasing Reproducibility in IR: Findings from the Dagstuhl Seminar on Reproducibility of Data-Oriented Experiments in e-Science. *ACM SIGIR Forum* 50, 1 (2016), 68–82.

[34] Nicola Ferro and Gianmaria Silvello. 2016. A General Linear Mixed Models Approach to Study System Component Effects. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '16)*. ACM, New York, NY, USA, 25–34. https://doi.org/10.1145/2911451.2911530

[35] Krishnendu Ghosh, Plaban Kumar Bhowmick, and Pawan Goyal. 2017. Using Re-ranking to Boost Deep Learning Based Community Question Retrieval. In *Proceedings of the International Conference on Web Intelligence (WI '17)*. ACM, New York, NY, USA, 807–814. https://doi.org/10.1145/3106426.3106442

[36] Mona Golestan Far, Scott Sanner, Mohamed Reda Bouadjenek, Gabriela Ferraro, and David Hawking. 2015. On Term Selection Techniques for Patent Prior Art Search. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. ACM, New York, NY, USA, 803–806. https://doi.org/10.1145/2766462.2767801

[37] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A Deep Relevance Matching Model for Ad-hoc Retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management (CIKM '16)*. ACM, New York, NY, USA, 55–64. https://doi.org/10.1145/2983323.2983769

[38] Suhit Gupta, Gail Kaiser, David Neistadt, and Peter Grimm. 2003. DOM-based Content Extraction of HTML Documents. In *Proceedings of the 12th International Conference on World Wide Web (WWW '03)*. ACM, New York, NY, USA, 207–214. https://doi.org/10.1145/775152.775182

[39] Zoltan Gyongyi and Hector Garcia-Molina. 2005. Web spam taxonomy. In *First international workshop on adversarial information retrieval on the web (AIRWeb 2005)*.

[40] A. Hanbury, G. Kazai, A. Rauber, and N. Fuhr (Eds.). 2015. *Advances in Information Retrieval: 37th European Conference on IR Research, ECIR 2015*. LNCS, Vol. 9022. Springer.

[41] Donna Harman and Chris Buckley. 2009. Overview of the Reliable Information Access Workshop. *Information Retrieval* 12, 6 (18 Jul 2009), 615–641. https://doi.org/10.1007/s10791-009-9101-4

[42] David Hawking. 2000. Overview of the TREC-9 Web Track. In *Proceedings of The Ninth Text REtrieval Conference, TREC 2000, Gaithersburg, Maryland, USA, November 13-16, 2000*. http://trec.nist.gov/pubs/trec9/papers/web9.pdf

[43] David Hawking and Nick Craswell. 2001. Overview of the TREC-10 Web Track. In *Proceedings of The Tenth Text REtrieval Conference, TREC 2001, Gaithersburg, Maryland, USA, November, 2001*.

[44] Hussein Hazimeh and ChengXiang Zhai. 2015. Axiomatic Analysis of Smoothing Methods in Language Models for Pseudo-Relevance Feedback. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval, ICTIR 2015, Northampton, Massachusetts, USA, September 27-30, 2015*. 141–150. https://doi.org/10.1145/2808194.2809471

[45] Ben He and Iadh Ounis. 2004. A Query-based Pre-retrieval Model Selection Approach to Information Retrieval. In *Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval (RIAO '04)*. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE, Paris, France, France, 706–719. http://dl.acm.org/citation.cfm?id=2816272.2816336

[46] Nasreen Abdul Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah S. Larkey, Xiaoyan Li, Mark D. Smucker, and Courtney Wade. 2004. UMass at TREC 2004: Novelty and HARD. In *Proc. TREC '04*.

[47] Wessel Kraaij and Thijs Westerveld. 2000. TNO-UT at TREC-9: How Different are Web Documents?. In *Proceedings of The Ninth Text REtrieval Conference, TREC 2000, Gaithersburg, Maryland, USA, November 13-16, 2000*. http://trec.nist.gov/pubs/trec9/papers/tno-ut.pdf

[48] Victor Lavrenko and W. Bruce Croft. 2001. Relevance Based Language Models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01)*. ACM, New York, NY, USA, 120–127. https://doi.org/10.1145/383952.383972

[49] Jimmy J. Lin, Matt Crane, Andrew Trotman, Jamie Callan, Ishan Chattopadhyaya, John Foley, Grant Ingersoll, Craig MacDonald, and Sebastiano Vigna. 2016. Toward Reproducible Baselines: The Open-Source IR Reproducibility Challenge. In *Advances in Information Retrieval - 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20-23, 2016. Proceedings*. 408–420. https://doi.org/10.1007/978-3-319-30671-1_30

[50] David E Losada and Leif Azzopardi. 2008. An analysis on document length retrieval trends in language modeling smoothing. *Information Retrieval* 11, 2 (2008), 109–138.

[51] Yuanhua Lv and ChengXiang Zhai. 2009. A Comparative Study of Methods for Estimating Query Language Models with Pseudo Feedback. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM '09)*. ACM, New York, NY, USA, 1895–1898. https://doi.org/10.1145/1645953.1646259

[52] Saptaditya Maiti, Deba P. Mandal, and Pabitra Mitra. 2011. Tackling Content Spamming with a Term Weighting Scheme. In *Proceedings of the 2011 Joint Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data*. ACM, New York, NY, USA, Article 6, 5 pages. https://doi.org/10.1145/2034617.2034624

[53] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. 2006. Detecting Spam Web Pages Through Content Analysis. In *Proceedings of the 15th International Conference on World Wide Web (WWW '06)*. ACM, New York, NY, USA, 83–92. https://doi.org/10.1145/1135777.1135794

[54] Jiaul H. Paik. 2015. A Probabilistic Model for Information Retrieval Based on Maximum Value Distribution. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. ACM, New York, NY, USA, 585–594. https://doi.org/10.1145/2766462.2767762

[55] Dae Hoon Park, Hyun Duk Kim, ChengXiang Zhai, and Lifan Guo. 2015. Retrieval of Relevant Opinion Sentences for New Products. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. ACM, New York, NY, USA, 393–402. https://doi.org/10.1145/2766462.2767748

[56] AFR Rahman, H Alam, and R Hartono. 2001. Content extraction from html documents. In *Proc. 1st Int. Workshop on Web Document Analysis (WDA2001)*. 1–4. http://wda2001.csc.liv.ac.uk//Papers/11_rahman_wda2001.pdf

[57] Fiana Raiber. 2012. Adversarial Content Manipulation Effects. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '12)*. ACM, New York, NY, USA, 993–993.

https://doi.org/10.1145/2348283.2348417

[58] Fiana Raiber and Oren Kurland. 2017. Kullback-Leibler Divergence Revisited. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval (ICTIR '17)*. ACM, New York, NY, USA, 117–124. https://doi.org/10.1145/3121050.3121062

[59] Fiana Raiber, Oren Kurland, and Moshe Tennenholtz. 2012. Content-based Relevance Estimation on the Web Using Inter-document Similarities. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*. ACM, New York, NY, USA, 1769–1773. https://doi.org/10.1145/2396761.2398514

[60] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* 3, 4 (April 2009), 333–389. https://doi.org/10.1561/1500000019

[61] S. E. Robertson and S. Walker. 1994. Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '94)*. Springer-Verlag New York, Inc., New York, NY, USA, 232–241. http://dl.acm.org/citation.cfm?id=188490.188561

[62] Gerard Salton and Chris Buckley. [n. d.]. SMART Stopword list. http://www.lextek.com/manuals/onix/stopwords2.html

[63] Amit Singhal, Chris Buckley, and Mandar Mitra. 1996. Pivoted document length normalization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 21–29.

[64] Amit Singhal, Gerard Salton, Mandar Mitra, and Chris Buckley. 1996. Document Length Normalization. *Inf. Process. Manage.* 32, 5 (1996), 619–633. https://doi.org/10.1016/0306-4573(96)00008-8

[65] Mark D. Smucker and James Allan. 2005. *An investigation of Dirichlet prior smoothing's performance advantage*. Technical Report. CIIR, U. Mass., Amherst.

[66] Ian Soboroff. 2013. Information retrieval evaluation demo. https://github.com/isoboroff/trec-demo

[67] Nikita Spirin and Jiawei Han. 2012. Survey on Web Spam Detection: Principles and Algorithms. *SIGKDD Explor. Newsl.* 13, 2 (May 2012), 50–64. https://doi.org/10.1145/2207243.2207252

[68] Fei Sun, Dandan Song, and Lejian Liao. 2011. DOM Based Content Extraction via Text Density. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '11)*. ACM, New York, NY, USA, 245–254. https://doi.org/10.1145/2009916.2009952

[69] Tim Weninger, William H. Hsu, and Jiawei Han. 2010. CETR: Content Extraction via Tag Ratios. In *Proceedings of the 19th International Conference on World Wide Web (WWW '10)*. ACM, New York, NY, USA, 971–980. https://doi.org/10.1145/1772690.1772789

[70] Craig Willis. 2017. Evaluation Framework - National Data Service - Confluence. https://opensource.ncsa.illinois.edu/confluence/display/NDS/Evaluation+Framework

[71] Chenyan Xiong and Jamie Callan. 2015. Query Expansion with Freebase. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval (ICTIR '15)*. ACM, New York, NY, USA, 111–120. https://doi.org/10.1145/2808194.2809446

[72] Peilin Yang and Hui Fang. 2016. A Reproducibility Study of Information Retrieval Models. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval (ICTIR '16)*. ACM, New York, NY, USA, 77–86. https://doi.org/10.1145/2970398.2970415

[73] Hamed Zamani and W. Bruce Croft. 2016. Estimating Embedding Vectors for Queries. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval (ICTIR '16)*. ACM, New York, NY, USA, 123–132. https://doi.org/10.1145/2970398.2970403

[74] Hamed Zamani and W. Bruce Croft. 2017. Relevance-based Word Embedding. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. ACM, New York, NY, USA, 505–514. https://doi.org/10.1145/3077136.3080831

[75] ChengXiang Zhai. 2008. *Statistical Language Models for Information Retrieval A Critical Review*. Vol. 2. Now Publishers Inc., Hanover, MA, USA. 137–213 pages. https://doi.org/10.1561/1500000008

[76] Chengxiang Zhai and John Lafferty. 2001. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proc. SIGIR*. ACM, New York, NY, USA, 334–342. https://doi.org/10.1145/383952.384019

[77] Chengxiang Zhai and John Lafferty. 2004. A Study of Smoothing Methods for Language Models Applied to Information Retrieval. *ACM Trans. Inf. Syst.* 22, 2 (April 2004), 179–214. https://doi.org/10.1145/984321.984322

[78] Guoqing Zheng and Jamie Callan. 2015. Learning to Reweight Terms with Distributed Representations. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '15)*. ACM, New York, NY, USA, 575–584. https://doi.org/10.1145/2766462.2767700