Passage language models in ad hoc document retrieval

Research Thesis

In Partial Fulfillment of The
Requirements for the Degree of
Master of Science in Information Management Engineering

Michael Bendersky

Submitted to the Senate of
the Technion - Israel Institute of Technology

Av, 5767 Haifa July 2007

The Research Thesis Was Done Under The Supervision of Dr. Oren Kurland in the Faculty of Industrial Engineering and Management.

The Generous Financial Help Of Technion Is Gratefully Acknowledged.

The work reported in this thesis is based upon work supported in part by the National Science Foundation under grant no. IIS-0329064. Any opinions, findings, and conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views or official policies, either expressed or implied, of any sponsoring institutions, the U.S. government, or any other entity.

Acknowledgements

I would like to thank my advisor Dr. Oren Kurland for his thoughtful guidance, his invaluable advice, and above all, his confidence in me throughout the work on this thesis. Oren's dedication to research, enthusiasm and expertise have been and will be a great source of inspiration for me.

I would like to express a deep gratitude to my parents, Lora and Yakov, for their constant love and support.

I would like to thank my dear wife Marina for standing by me when I needed her the most, and for her wise advice, understanding and unconditional love.

Contents

List of Figures

Abstract

Throughout the last decade or so, search engines (e.g., Google) have become a crucial tool for discovering information in on-line data repositories. Finding documents in a *corpus* (repository) that pertain to users' queries is a hard challenge, especially in light of increasingly large and diverse corpora as the World Wide Web (WWW), for example.

The *ad hoc retrieval* task, which is the principle task that search engines perform, is to rank documents in a corpus in response to a query by their assumed relevance to the information need it represents. While there are numerous challenges involved in ad hoc retrieval, one of the prominent ones is the fact that a document could be of interest to a user (or deemed relevant to the user's query) even if only (very few, potentially small) parts of it, i.e., *passages*, actually contain information that pertains to the information need. Therefore, in such situations methods that compare the document as a whole to the query face significant difficulties in detecting the required documents.

Passage-based document-retrieval approaches address this challenge by using the information from (only some of) the document passages to rank a document in response to a query. In this thesis we show that several of these previously proposed passage-based approaches, along with some new ones, can in fact be derived from the same probabilistic model.

While our formulation and derived ranking algorithms are not committed to any specific estimation paradigm, we use the successful *language modeling* framework to instantiate specific algorithms. In doing so, we propose a novel *passage language model* that integrates information from the ambient document to an extent controlled by the estimated *document homogeneity*. Several document homogeneity measures that we propose yield passage language models that are more effective than previously proposed ones. Furthermore, we demonstrate the benefits in using our proposed passage language model for constructing and utilizing a passage-based *relevance model*.

Finally, we show that our proposed document-homogeneity measures are also effective means for integrating document-query and passage-query similarity information for document retrieval.

1 Introduction

With the enormous increase in recent years in the volume of information available on-line, and the consequent need for better techniques to access this information, there has been a strong resurgence of interest in information retrieval research.

Search engines, e.g., Google[1], play an important role nowadays in tasks such as information discovery and retrieval. A search engine lets a user submit a query representing some information need and retrieves in response a list of documents that are considered relevant to this need. This list is often sorted with respect to some measure of assumed relevance of the results. Thus, search engines perform the *ad hoc retrieval task*, which is to rank documents in response to a query by their assumed relevance to the information need it represents [42]. Search engines offer an access to an unprecedented amount of heterogeneous and unstructured information, and selecting and ranking specific documents by their relevance to users' queries from such large and diverse corpora is a hard challenge.

While there are numerous challenges involved in ad hoc document retrieval, one of the prominent ones is the fact that a document could be of interest to a user (or deemed relevant to a user's query under certain relevance-judgment regimes) even if only (very few, potentially small) parts of it, i.e., *passages*, actually contain information pertaining to the query. For example, consider a web page containing messages from various news feeds. It could be the case that a single feed is relevant to a certain query; however, if we treat the web page as a single monolithic document this relevance could potentially have only limited influence on the overall page ranking in response to the query. On the other hand, if we treat the web page as composed of passages (passages in this case are the various feeds), we could use this view in order to better match the document and the query (e.g., show the user the most relevant feed in response to her query).

Indeed, passage identification and utilization in information retrieval has been the focus of research for quite some time [19, 7, 45, 28]. Utilization of passages has been shown to be highly beneficial for a variety of information retrieval tasks: classical ad hoc retrieval [7, 28, 19, 45, 6], question answering [15, 8], query expansion [5] and classification [10] to name just a few. In the case of ad hoc retrieval, one could either choose to return the relevant passages as a result [2], or to simply mark the entire document as relevant if it contains (some) relevant passage(s) [7, 28, 45].

[1] http://www.google.com/

The focus of the work presented here is on the latter — deriving passage-based methods for retrieving documents. Indeed, the merits of passage-based document retrieval have long been recognized [37, 7, 31, 45, 18, 10, 19, 28]. Perhaps the most prominent one is that using passages rather than whole documents to induce document ranking is more effective for detecting long (or *heterogeneous*) relevant documents with many parts that contain no query-relevant information, as in the case of the web page example from above.

In this thesis we present a simple formal probabilistic formulation for passage-based ad hoc document retrieval. Using this formulation we show that some previously proposed passage-based document retrieval principles [7, 45, 19, 28] can be derived from the same model if some assumptions and estimation choices are made. We present several concrete instantiations of our probabilistic formulation.

One such instantiation ranks a document by the highest query-similarity score of any of its passages, which echoes some past work [7, 45, 19, 28]; another instantiation interpolates this score with the document-query similarity score [7, 45], which also bears similarity to some previously proposed approaches [7, 45, 6]. We also derive a generalized form of the latter by controlling the reliance on document-based versus passage-based query-similarity evidence using *document homogeneity* measures, which we propose: the more heterogeneous the document is assumed to be, the more weight is given to passage-based evidence.

Our formulation and derived methods are not committed to a specific estimation paradigm. To instantiate specific algorithms, however, we choose the successful *language modeling* framework to retrieval [34, 9]. In doing so, we derive a new *passage language model* that utilizes information from the ambient document to an extent controlled by the same homogeneity measures used for the interpolation-based ranking approach from above; the more heterogeneous the ambient document is considered to be, the less the passage language model relies on information from other passages in the document.

Using an array of experiments performed over various TREC corpora, we show that few of the document-homogeneity measures that we propose yield passage language models that are more effective than the standard passage model [28] for basic passage-based document ranking and for constructing and utilizing *passage-based relevance models* [28]. The later also outperform a document-based relevance model [25].

Furthermore, we explore the retrieval performance of a novel language-model-based algorithm that integrates document-query and passage-query

similarity information based on the proposed document-homogeneity measures. Experimental results demonstrate the effectiveness of this algorithm with respect to standard document-based and passage-based document retrieval in the language modeling framework.

Finally, we further demonstrate the merits in using document-homogeneity measures through a comparison with the common practice of fixing the balance in utilizing document versus passage information to the same degree for all documents [7, 1, 15, 32, 44].

The remainder of this thesis is organized as follows. Chapter 2 lays down the basic concepts of the language modeling approach to information retrieval that are used throughout this thesis. Chapter 3 surveys the related work on passages' identification and utilization in information retrieval in general and specifically in the context of language models. Chapter 4 presents the probabilistic formulation of the passage-based ad hoc document retrieval task, while Chapter 5 presents our novel homogeneity-based passage language model that is used to instantiate some specific retrieval methods. Chapter 6 presents a view of some of our passage-based retrieval models as standard document retrieval methods that utilize a certain form of document representation. Chapter 7 describes the various experiments we conducted to test the performance of our retrieval methods. In Chapter 8 we draw conclusions and discuss some potential directions for future work.

2 Statistical Language Models

Statistical language models play a central role in this work. In this chapter we overview their utilization in ad hoc information retrieval and lay out the definitions and notations that will be used throughout this thesis.

A statistical language model is a probability distribution that captures the statistical regularities of language generation [35]. It determines how likely a given string is in a language, given a model of language generation. Statistical language modeling is used in a large variety of language technology applications. These include speech recognition, machine translation, document classification and routing, optical character recognition, handwriting recognition, spelling correction, information retrieval and many more. (See Rosenfeld [35] for a survey of language models and their use in various fields)

There is an abundance of work on application of language models in information retrieval, since their first use by Ponte and Croft [34]; among the most frequently used models is the *query-likelihood model* [34, 30, 13, 40].

In the query-likelihood model, one estimates the probability of a query being generated by a probabilistic distribution over a fixed vocabulary induced by a document. For a query q and a document d this generation probability is often denoted $p(q|d)$. In order to rank documents we use the posterior probability $p(d|q)$ [34, 24], which can be written using Bayes' rule as

$$p(d|q) = \frac{p(q|d)p(d)}{p(q)}.$$

Since $p(q)$ is not dependent on the document and in lack of prior information $p(d)$ is assumed to be uniformly distributed, the ranking task reduces to estimating $p(q|d)$. This model has been commonly used in work on language models in information retrieval [34, 25, 46, 40, 13, 30] (see Lafferty and Zhai [24] for extended details), and will be used in this thesis as well.

2.1 Unigram language models

While there is a large number of methods for estimating the probability $p(q|d)$, in this thesis we take the widely used approach in ad hoc retrieval and utilize *unigram language models*, which were shown to be quite effective [9, 40, 46, 30]. Unigram language models assume that terms are independent of each other. We use $p_x(\cdot)$ to denote the (smoothed) unigram language

model induced from text x. Next, we present an approach for estimating $p_x(\cdot)$.

Language model induction Let $\mathrm{tf}(w \in x)$ denote the number of occurrences of term w in the text x. The maximum likelihood estimate (MLE) of w with respect to x is

$$\widetilde{p}_x^{MLE}(w) \stackrel{def}{=} \frac{\mathrm{tf}(w \in x)}{\sum_{w'} \mathrm{tf}(w' \in x)}.$$

To assign probability to terms unseen in x (a.k.a. the *zero probability problem*), we smooth the estimate using corpus statistics [46]

$$\widetilde{p}_x^{[basic]}(w) = (1 - \lambda_{\mathcal{C}})\widetilde{p}_x^{MLE}(w) + \lambda_{\mathcal{C}}\widetilde{p}_{\mathcal{C}}^{MLE}(w). \quad (1)$$

In the above estimate $\lambda_{\mathcal{C}}$ is a free parameter. Setting $\lambda_{\mathcal{C}}$ to a fixed value, we get the Jelinek-Mercer smoothing technique [46]. Alternatively, we can set $\lambda_{\mathcal{C}} = \frac{\mu}{|x|+\mu}$ ($|x| = \sum_{w'} \mathrm{tf}(w' \in x)$; μ is a free parameter) and get the Bayesian smoothing approach with Dirichlet priors [46].

To extend the estimate $\widetilde{p}_x^{[basic]}(w)$ to a sequence of terms $w_1 w_2 \cdots w_n$, we follow the unigram-language-model term-independence assumption and define

$$p_x^{[basic]}(w_1 w_2 \cdots w_n) \stackrel{def}{=} \prod_{j=1}^{n} \widetilde{p}_x(w_j). \quad (2)$$

We can use the estimate just defined for estimating $p_x(\cdot)$.

2.2 Relevance models

In the following chapters we will also examine the utilization of passages for constructing relevance models [28]. We now describe the fundamentals of the relevance model approach [25].

Lavrenko and Croft [25] take the following generative perspective on the ad hoc retrieval task. They make the assumption that both the query and the relevant documents are samples from an underlying *relevance model* $\mathcal{R}$. In order to estimate the relevance model they use the joint probability of observing some term w together with the terms $q_1, \ldots, q_m$ of query q

$$\widetilde{p}_{\mathcal{R}}(w) \approx \frac{p(w, q_1, \ldots, q_m)}{p(q_1, \ldots, q_m)}. \quad (3)$$

Lavrenko and Croft [25] describe two methods of estimating the joint probability $p(w, q_1, \ldots, q_m)$. Both methods assume that there exists a set $\mathcal{D}$ of underlying source distributions from which w and q's terms could have been sampled. The two methods differ in the independence assumptions they make. Method 1 assumes that w and q's terms are mutually independent once we pick a source distribution from $\mathcal{D}$. Method 2 assumes that q's terms are independent of each other, but are dependent on the choice of w. Following prior work on passage language models [28], and given the general superiority of Method 1 to Method 2 [26] we utilize Method 1 in our work and present it formally in what follows.

If we set $\mathcal{D}$ to be the set of all document models and assume mutual independence of w and q's terms $q_1, \ldots, q_m$, we can write

$$p(w, q_1, \ldots, q_m) = \sum_{d \in \mathcal{D}} p(d)p(w|d) \prod_{i=1}^{m} p(q_i|d) = \sum_{d \in \mathcal{D}} p(d)p(w|d)p(q|d).$$

Using Bayes' rule, this can be rewritten as

$$p(w, q_1, \ldots, q_m) = \sum_{d \in \mathcal{D}} p(q)p(w|d)p(d|q).$$

If we substitute the equation above into Equation 3, $p_{\mathcal{R}}(w)$ is then

$$\widetilde{p}_{\mathcal{R}}(w) \approx \sum_{d \in \mathcal{D}} p(d|q)p(w|d),$$

where $p(d|q)$ can be estimated by

$$p(d|q) = \frac{p(q|d)p(d)}{p(q)} = \frac{p(q|d)p(d)}{\sum_{d \in \mathcal{D}} p(d)p(q|d)}.$$

In absence of any prior information $p(d)$ can be assumed to be uniformly distributed. $p(w|d)$ and $p(q|d)$ can be estimated using $p_d(\cdot)$, the smoothed unigram language model induced from d and calculated using Equation 1, wherein x corresponds to document d from $\mathcal{D}$. Note that since in practice $p_d(q)$ will be near-zero for all but a few highest-scoring documents in the collection, we can compute $p_{\mathcal{R}}(w)$ using only the top-n retrieved documents in a search performed using $p(q|d)$ (i.e., the query likelihood model) [26]. This allows the estimation process to scale well for large corpora.

After the estimation of $p_{\mathcal{R}}(\cdot)$ is accomplished, following Lavrenko et al. [26], we use the Kullback-Leibler divergence metric to rank documents.

Specifically, we use the KL divergence between a relevance model $\mathcal{R}$ and a document model $p_d(\cdot)$, which is defined as:

$$D\left(\widetilde{p}_{\mathcal{R}}(\cdot)\;\middle|\middle|\;\widetilde{p}_d(\cdot)\right) = \sum_w \widetilde{p}_{\mathcal{R}}(w) \log \frac{\widetilde{p}_{\mathcal{R}}(w)}{\widetilde{p}_d(w)}. \tag{4}$$

Documents are ranked in increasing divergence order, i.e., documents that have a smaller divergence from the relevance model are considered to be more relevant to the query.

3 Related Work

The inspiration for the work reported in this thesis is drawn from two areas of research in information retrieval: language modeling and passage retrieval. Language models are discussed in detail in Chapter 2. In this chapter we give an overview of past research on passage retrieval in Section 3.1 and describe the work done on utilization of passage language models in Section 3.2.

3.1 Passage retrieval

The ad hoc retrieval task is to find information pertaining to a need expressed by some query. Perhaps the most well known form of ad-hoc retrieval is document retrieval. However, there are cases wherein documents are highly heterogeneous, and using only the most relevant document portions might be of value. We refer to these portions as *passages*.

Passages can be used in two ways for ad hoc retrieval. First, we can return passages as a result of the query. Alternatively, passages can be used to retrieve documents. In both cases, the retrieval task is to find passages that might pertain to a user's query. In the second case, however, these passages are used to evaluate the relevance of their ambient documents. The focus of this thesis is on the latter.

As described above, passage-based document retrieval can be of help in cases wherein only small portions of a relevant document contain information that is relevant to the query. In such cases, when metrics that compare the entire document to the query are computed for the purpose of document ranking, the non-relevant document parts potentially mask the relevant passages' contribution to the overall score [45, 7, 11]. For example, consider a comprehensive book on the topic of information retrieval, wherein only a single section discusses passage-based retrieval [11]. If the entire book is considered as an indivisible monolithic document, this section will have very limited influence on the overall document rank for a query discussing the subject of passage-based retrieval.

The main challenges in passage-based document retrieval research are the identification of passage boundaries, the detection of relevant passages in response to the query and the combination of passage-query and document-query similarity scores. In the following sections we will discuss these challenges and survey the use of passages in various tasks in information retrieval.

3.1.1 Passage identification

Passage types can be roughly classified into three main groups [7, 19]: *discourse passages, semantic passages* and *window passages*.

Discourse passages are based on the document markup; examples include sentences, paragraphs or sections boundaries. Discourse passages have been found to work well for highly structured and edited corpora with clearly defined boundaries (e.g., encyclopedia text [38], SGML-tagged text data as in the AQUAINT Corpus of English News Text[2] [15] and HTML mark-up [6]). However, in more heterogeneous collections where mark-up is less rigid and document length and structure exhibit significant variations, discourse passages do not seem to result in consistent retrieval performance [7].

Semantic passages are based on shifts of topic within a document. One of the efficient techniques to derive semantic passages is TextTiling [12]. This technique groups adjacent blocks of text with high similarity into passages. Blocks are derived from sentence punctuation, and the similarity measure is the cosine between the vector-space representation of pairs of adjacent blocks. Among other methods proposed for semantic passage identification are text segmentation using the LCA method [33] and Hidden Markov Models [31, 10].

Window passages are passages that are based on fixed (or variable) number of words. This simple passaging technique was shown in some cases to be at least as effective as other techniques for passage identification for document retrieval [7, 18, 28]. This can be explained by the fact that semantic or structural features may be hard to identify in heterogeneous corpora [18]. A possible problem with dividing text into disjoint windows is that a small block of relevant text may be split between two passages. To overcome this problem overlapping windows are often used [7, 11]. Callan [7] proposes the following approach for building overlapping windows: begin the first passage in a document at the first term matching the query and create a new passage of length n every $\frac{n}{2}$ words. Liu and Croft [28] propose a similar method, except that the first passage begins at the first word of the document. An important difference between these two passaging methods is that the former is query-dependent, i.e., passages are built at retrieval time, while the latter is query-independent, and passages can be built off line. In our work, the latter approach was adopted, since it results in improved retrieval runtime, as passage discovery and indexing is performed prior to retrieval time. In addition, this type of query-independent passages was shown to be quite effective, especially in the context of language models [28, 44].

[2] http://www.ldc.upenn.edu/Catalog/docs/LDC2002T31/

Kaszkiel and Zobel [18, 19] propose a more robust approach for building window passages that they term as *arbitrary passages*. Instead of considering a single passage size, arbitrary passages are built using text segments of varying lengths. This assures that the retrieval effectiveness is not hindered by selection of an unappropriate passage size. When no resource constraints are imposed, arbitrary passages are all text segments of every possible length starting at every word in a document. Since this results in a very large number of possible passages, several approximations are proposed, such as restricting possible passage lengths to certain values [18]. Although retrieval using arbitrary passages was shown to be quite effective [18, 19, 28], it requires a considerable computation overhead, both due to the large number of potential passages, and the fact that the passages are calculated dynamically at query time [19].

3.1.2 Utilizing passages in Information Retrieval

Passage utilization has several important advantages in information retrieval context. First, since passages are short, they embody locality of information — if several query terms appear in a single passage, they must be close to one another, which potentially implies higher relevance [18, 7]. Second, document passages can serve as document *previews* that enable quick location of relevant document portions by the users [44]. Third, passage retrieval can be used as an intermediate step for summarization and question answering systems [15, 32, 8].

Indeed, passages have been used extensively for a variety of tasks in information retrieval. One frequent use of passages is for question answering systems [8, 47, 15]. In such systems, passages serve as an intermediary between full documents and exact answers, and almost all question answering systems implement a technique for extracting passage fragments of text from a large corpus [41]. Some recent user studies [27] have shown that users often prefer passage-sized chunks of text over exact phrase answers returned in response to their queries, because the former provide context.

Another possible use of passages is in sentence or passage retrieval [10, 16, 32, 44]. In this type of retrieval, most relevant document passages (or sentences) are returned to the user in response to a query, instead of documents.

Passages have also been used in web-retrieval [6], where web pages were segmented into passages using HTML mark-up; query expansion [5], where top-ranked passages were used for expanding the query instead of documents; and document retrieval [37, 7, 45, 31, 10, 28], where passage query-

similarity evidence was used for document ranking.

A combination of evidences from both the highest-scoring document passage and the document itself (scores represent similarity to a query or question) was shown to be beneficial for many of the above tasks [7, 45, 6, 15, 32, 44]. This strengthens the hypothesis (see Chapter 1) that *both* passage context (e.g., in case of a single relevant passage of highly heterogeneous document) and the ambient document context (e.g., in case of a highly homogeneous document) might play a vital role in determining the relevance of a document.

There is an abundance of methods that utilize passage-query similarity information for document retrieval. These include: interpolation of evidence from the passage most similar to the query with document-query similarity evidence using fixed weights [5, 7, 6, 45], using the highest scoring passage to rank documents [18, 19, 28] and utilizing the weighted sum of k highest ranked passages to rank a document [45]. We show in Chapter 4 that many of these ranking approaches can be derived (and generalized) from the same model.

3.2 Passage-based retrieval utilizing language models

Since passages are spans of text, techniques presented in Chapter 2 for language model induction can be naturally employed to construct passage language models.

One task for which utilization of passage language models was shown to be useful is question answering [8, 47, 15]. As mentioned in Section 3.1.2, passage retrieval and ranking is an essential part of a typical question answering system. Corrada-Emmanuel et al. [8] rank the retrieved passages using the Kullback-Leibler divergence between the *relevance model* [25] and a passage model (this approach is similar to the original relevance model approach, except that the relevance model is built from the top ranked passages rather than from top ranked documents). Corrada-Emmanuel et al. [8] also show that using passage-based relevance model often results in improved performance over the passage-based query-likelihood model [34]. In both cases, passage models are smoothed by the collection statistics.

Hussain [15] takes an approach more similar to the one we will describe in Section 5 in that whole document evidence is combined with passage evidence when ranking passages in a question-answering system. Passages are first ranked using basic passage-based query-likelihood model. The top ranked passages are then re-ranked using several techniques. The largest performance improvement over the passage-based query-likelihood is ob-

tained using a model that interpolates between the language model of a passage and the language model of its ambient document. Optimal interpolation parameters' values are determined using an exhaustive search over the entire parameter space.

Similarly to Hussain's approach, some work on sentence and passage retrieval [1, 44, 32] interpolate the document fragment model with its ambient document's statistics. Murdock and Croft [32] present two approaches for interpolation in sentence retrieval. The first approach smoothes the sentence language model with statistics pooled from the surrounding context backed-off by the whole document language model. The second approach smoothes sentence, document and collection language models using Jelinek-Mercer smoothing [46]. Both approaches significantly outperform the baseline, which is a query-likelihood model based on a sentence language model smoothed with collection statistics. The second approach demonstrates the best performance among the two. Wade and Allan [44] compare several passage retrieval methods, and their *Mixture of Language Models* method, which smoothes the document language model with the passage language model and the collection language models (albeit for passage retrieval, rather than for document retrieval), was shown to be the most effective one.

In all of the above mentioned cases, interpolation of document and passage evidence was based on fixed weights. In contrast, we will show in Chapter 4 how to automatically set these weights based on several document properties.

Liu and Croft's work [28] on passage-based-language-model document retrieval most resembles ours in that they use the passage with the highest query-similarity to rank its ambient document. In addition, they rank documents using a passage-based *relevance model* [25]. We compare the performance of their method to ours in Chapter 7.

4 Retrieval Framework

In this chapter we show that a few previously proposed passage-based document ranking principles, along with some new ones, can be derived from the same probabilistic model.

Throughout this chapter we assume that the following have been fixed: a query q, a document d, which we want to score in response to q, and a static corpus of documents $\mathcal{C}$ (to which d belongs). We assume that passages are identified for each document in the corpus either *before* or *during* retrieval time. (Our formulation and derived algorithms are independent of the type of passages being used.) We denote a passage as g, and write $g \in d$ if g is part of d; we assume that d has m passages.

4.1 Ranking models

In the ad hoc retrieval setting, the goal is to rank documents in response to a query. To that end, we take a probabilistic approach [34, 24], and score d in response to q by $p(d|q)$. Assuming uniform prior distribution for documents, our task reduces to estimating $p(q|d)$, which in the language model framework [9], for example, can be interpreted as $p_d(q)$ — the probability assigned to q by a language model induced by d. (See Chapter 2 for elaborated discussion on language models.) We hasten to point out, however, that our formulation and derived ranking models in this chapter are not committed to any specific estimation paradigm.

Now, given the passages in the corpus, we use basic probability theory and write

$$p(q|d) = \sum_{g_i} p(q|d, g_i)p(g_i|d), \tag{5}$$

where g_i is *some* passage of *some* document in the corpus.

While we can (theoretically) use Equation 5 to rank d using all passages in the corpus, our goal here is to score it using only its own passages. Thus, we assume an estimate $\hat{p}(g_i|d)$ — which we can interpret as "how good a representative is g_i of d" — for which $\sum_{g_i \in d} \hat{p}(g_i|d) = 1$ holds.

We can derive $\hat{p}(\cdot|d)$, for example, given a model of $p(\cdot|d)$ by setting

$$\hat{p}(g_i|d) \stackrel{def}{=} \delta[g_i \in d] \frac{p(g_i|d)}{\sum_{g_j \in d} p(g_j|d)}$$

($\delta[g_i \in d] = 1$ if and only if $g_i \in d$; note that $\sum_{g_i \in d} \hat{p}(g_i|d) = 1$ holds).

We can then use the estimate $\hat{p}(g_i|d)$ in Equation 5, and get our basic passage-based document scoring function

$$Score(d) \stackrel{def}{=} \sum_{g_i \in d} p(q|d, g_i)\hat{p}(g_i|d). \quad (6)$$

If we "believe" that a document should be scored by the "matches" of (some of) its passages to the query regardless of its "match" as a whole to the query — recalling the observation in Chapter 1 regarding long and heterogeneous documents — we can then make the assumption that a *query is independent of a document given a passage* and get

$$Score(d) \stackrel{def}{=} \sum_{g_i \in d} p(q|g_i)\hat{p}(g_i|d). \quad (7)$$

Note that if d's passages are marked-up sections with varying degrees of importance, we can use $\hat{p}(g_i|d)$ as an estimate for this importance and get that Equation 7 is one of Wilkinson's better performing models [45].

In lack of any such additional information regarding passages' relative importance, we might make the assumption that d's passages are all equal representatives of d and use uniform distribution for $\hat{p}(g_i|d)$. Equation 7 then reduces to scoring d by the mean score of its constituent passages

$$Score_{mean}(d) \stackrel{def}{=} \frac{1}{m} \sum_{g_i \in d} p(q|g_i). \quad (8)$$

However, this ranking criterion implies that many passages in d should contain information pertaining to q for d to be considered relevant, in contrast to our goal from Chapter 1 of detecting also long (heterogeneous) documents that might contain a single relevant passage.

Instead of assuming a uniform distribution for $\hat{p}(g_i|d)$, we can use $\max_{g_i \in d} p(q|g_i)$ and the fact that $\sum_{g_i \in d} \hat{p}(g_i|d) = 1$ to derive a bound for the score in Equation 7

$$Score_{max}(d) \stackrel{def}{=} \max_{g_i \in d} p(q|g_i); \quad (9)$$

this scoring function was used in some previous work on passage-based document ranking [7, 19, 45, 28].

The ranking models in Equations 7, 8 and 9 score a document only by (some of) its passages' "matches" to the query. We now consider the alternative of combining this information with the "match" of the document as whole to the query, wherein we balance the two sources of information by the

assumed *homogeneity* level of the document — i.e., the more homogeneous the document is, the higher the impact its "match" to the query has on the final score. To achieve that, we first drop the independence assumption from above ("a query is independent of a document given a passage") and use the estimate[3]

$$\hat{p}(q|d, g_i) \stackrel{def}{=} h^{[\mathcal{M}]}(d)p(q|d) + (1 - h^{[\mathcal{M}]}(d))p(q|g_i),$$

where $h^{[\mathcal{M}]}(d)$ assigns a value in $[0, 1]$ to d according to a homogeneity model $\mathcal{M}$. (Higher values of $h^{[\mathcal{M}]}(d)$ represent higher estimates of homogeneity; we present various document-homogeneity measures in Section 4.2.)

Using this estimate in Equation 6 we get

$$Score_{inter}(d) \stackrel{def}{=} h^{[\mathcal{M}]}(d) \sum_{g_i \in d} p(q|d)\hat{p}(g_i|d) + (1 - h^{[\mathcal{M}]}(d)) \sum_{g_i \in d} p(q|g_i)\hat{p}(g_i|d).$$

Recalling that $\sum_{g_i \in d} \hat{p}(g_i|d) = 1$ the equation above reduces to

$$Score_{inter}(d) = h^{[\mathcal{M}]}(d)p(q|d) + (1 - h^{[\mathcal{M}]}(d)) \sum_{g_i \in d} p(q|g_i)\hat{p}(g_i|d).$$

Using the observations from above, we can now bound this score by

$$Score_{inter-max}(d) \stackrel{def}{=} h^{[\mathcal{M}]}(d)p(q|d) + (1 - h^{[\mathcal{M}]}(d)) \max_{g_i \in d} p(q|g_i), \qquad (10)$$

which puts more weight on document-based evidence as d is estimated to be more homogeneous. Indeed, setting $h^{[\mathcal{M}]}(d) = 1$, assuming d is highly homogeneous, we get a document-based ranking approach. On the other hand, assuming that d is extremely heterogeneous and setting $h^{[\mathcal{M}]}(d) = 0$, we score d as in Equation 9 that depends only on d's passages.

Note that the scoring function in Equation 10 is a generalization of past approaches [5, 7, 6, 45] wherein a document score and the maximal score that any of its passages is assigned are interpolated using fixed weights.

While the scoring functions from above can be instantiated using different estimates for $p(q|d)$ and $p(q|g_i)$, we follow the effective language modeling approach to IR and utilize language models, as will be described in Chapter 5.

[3]This is reminiscent of some recent work on cluster-based retrieval [21].

4.2 Document homogeneity

We now consider a few simple models $\mathcal{M}$ for estimating document d's homogeneity, i.e., we define functions $h^{[\mathcal{M}]} : \mathcal{C} \rightarrow [0, 1]$ with higher values corresponding to (assumed) higher levels of homogeneity.

Long documents have often been considered as more heterogeneous than shorter ones [39]. Intuitively, the chances for content heterogeneity in a document increase as the number of the terms it contains grows. We define document length $|d|$ as $\sum_{w'} \mathrm{tf}(w' \in d)$ (the number of terms it contains), and formulate a normalized length-based measure with respect to the longest document in the corpus[4]

$$h^{[length]}(d) \stackrel{def}{=} 1 - \frac{\log |d| - \min_{d_i \in \mathcal{C}} \log |d_i|}{\max_{d_i \in \mathcal{C}} \log |d_i| - \min_{d_i \in \mathcal{C}} \log |d_i|}.$$

However, the length-based measure just described does not handle the case of short heterogeneous documents. We can alternatively say that d is more homogeneous if its term distribution is concentrated around a small number of terms [22]. To model this idea, we use the normalized *entropy* of $d's$ unsmoothed language model. Document entropy is defined as

$$Entropy(d) = -\sum_{w' \in d} \widetilde{p}_d^{MLE}(w') \log \widetilde{p}_d^{MLE}(w')$$

(higher values correspond to (assumed) lower levels of homogeneity). We then normalize the measure with respect to the maximum possible entropy of a document with the same length as that of d's , that is, a document in which each term is repeated exactly once (i.e., $Entropy(d) = \log |d|$). Thus, the entropy-based measure is

$$h^{[ent]}(d) \stackrel{def}{=} \begin{cases} 1 + \frac{\sum_{w' \in d} \widetilde{p}_d^{MLE}(w') \log(\widetilde{p}_d^{MLE}(w'))}{\log |d|} & |d| > 1 \\ 1 & \text{otherwise} \end{cases}$$

Both homogeneity measures just described are based on the document as a whole and do not explicitly estimate the variety among its passages. We can say, for example, that the more similar the passages of a document are to each other, the more homogeneous the document is. Alternatively, a

[4]Note that this measure is corpus-dependent — the same document may have different homogeneity values across different corpora. Normalization with respect to the longest document in all tested corpora yielded similar results to those of the original proposal defined here.

document with passages highly similar to the document as a whole might be considered homogeneous.

To formally capture these two homogeneity notions, we assume that the passages of d are assigned with unique IDs, and denote the tf.idf[5] vector-space representation of text x as $\vec{x}$ [36] ; we can then define these notions using

$$h^{[interPsg]}(d) \stackrel{def}{=} \begin{cases} \frac{2}{m(m-1)} \sum_{i<j;g_i,g_j \in d} \cos(\vec{g_i}, \vec{g_j}) & \text{if } m > 1 \\ 1 & \text{otherwise} \end{cases}$$

and

$$h^{[docPsg]}(d) \stackrel{def}{=} \frac{1}{m} \sum_{g_i \in d} \cos(\vec{d}, \vec{g_i}),$$

respectively.

Although it is clear that the document homogeneity measures $h^{[interPsg]}(d)$ and $h^{[docPsg]}(d)$ are connected, they differ in their sensitivity to passages with content strongly deviating from the content of the rest of the passages in the document. Case in point, measure $h^{[docPsg]}(d)$ is more *conservative* than the $h^{[interPsg]}(d)$ measure in estimation of the document homogeneity. For example, consider a document containing two passages g_1 and g_2, such that $g_1 \cap g_2 = \emptyset$. Referring to the document homogeneity measures as defined above, we get that $h^{[interPsg]}(d) = 0$ (as passages' contents are disjoint), while $h^{[docPsg]}(d) > 0$ (as each passage bears some similarity to the document as a whole).

[5]Modeling the latter two homogeneity notions utilizing a (normalized) version of the KL-divergence between language models yielded retrieval performance substantially inferior to that resulting from using a vector space representation with the cosine measure.

5 Language Model Framework

In the previous chapter we showed that a few previously proposed passage-based document ranking models, along with some new ones, can be derived from the probabilistic model we presented. While the ranking models we derived can be instantiated using different estimates for $p(q|d)$ and $p(q|g_i)$, we follow the effective language modeling approach to IR and utilize language models to instantiate specific algorithms; specifically, we use language models for estimating these quantities.

Following standard practice in work on language models for IR [9], we can estimate $p(q|d)$ and $p(q|g_i)$ using the unigram language models $p_d(q)$ and $p_{g_i}(q)$ respectively (see Chapter 2, page 4). Thus, we get the following algorithms:

The Mean Passage-Scoring algorithm assumes uniform distribution for $\hat{p}(g_i|d)$ and scores d by

$$\frac{1}{m}\sum_{g_i \in d} p_{g_i}(q),$$

(see Equation 8, page 14).

The Max-Scoring Passage algorithm suggested by Liu and Croft [28] and scores d by

$$\max_{g_i \in d} p_{g_i}(q),$$

(see Equation 9, page 14).

The Interpolated Max-Scoring Passage algorithm is a novel algorithm, which scores d by

$$h^{[\mathcal{M}]}(d)p_d(q) + (1 - h^{[\mathcal{M}]}(d)) \max_{g_i \in d} p_{g_i}(q),$$

(see Equation 10, page 15).

Recall from Chapter 2 the standard language unigram language model induced from text x

$$p_x^{[basic]}(w_1 w_2 \cdots w_n) \stackrel{def}{=} \prod_{j=1}^{n} \widetilde{p}_x^{[basic]}(w_j)$$

We can use this basic language model to estimate $p_d(q)$ and $p_{g_i}(q)$ in our algorithms.

Using $p_{g_i}^{[basic]}(q)$, however, implies that *every* document is so heterogeneous that in scoring each of its passages we do not consider any information from d, except for that in the passage itself. We would, however, want to leverage information from d to estimate its passages' "matches" to the query to an extent controlled by d's estimated homogeneity. Note that this practice does *not* result in the Max-Scoring Passage algorithm being equivalent to the Interpolated Max-Scoring Passage algorithm, since the latter integrates document-based information only *after* the maximal-scoring passage has been determined.

Inspired by some past work [1, 15, 32, 44], we define a passage language model that exploits information from the ambient document using interpolation-based smoothing. In contrast to this past work, however, wherein the interpolation is based on fixed weights that control the amount of reliance on document statistics, we adopt the underlying concept of our Interpolated Max-Scoring Passage algorithm (that performs interpolation at the score level), and control this reliance by the document estimated homogeneity as induced by model $\mathcal{M}$. We thus define the following term-based passage language model for $g \in d$

$$\widetilde{p}_g^{[\mathcal{M}]}(w) \stackrel{def}{=} \lambda_{psg}(g)\widetilde{p}_g^{MLE}(w) + \lambda_{doc}(d)\widetilde{p}_d^{MLE}(w) + \lambda_{\mathcal{C}}\widetilde{p}_{\mathcal{C}}^{MLE}(w), \qquad (11)$$

where we fix $\lambda_{\mathcal{C}}$ to some value; we then set $\lambda_{doc}(d) = (1 - \lambda_{\mathcal{C}})h^{[\mathcal{M}]}(d)$ and to ensure proper probability distribution we set $\lambda_{psg}(g) = 1 - \lambda_{\mathcal{C}} - \lambda_{doc}(d)$. We extend this estimate to sequences as in Chapter 2

$$p_{g\in d}^{[\mathcal{M}]}(w_1 w_2 \cdots w_n) \stackrel{def}{=} \prod_{j=1}^{n} \widetilde{p}_{g\in d}^{[\mathcal{M}]}(w_j). \qquad (12)$$

We observe that if we set $h^{[\mathcal{M}]}(d) = 0$ (i.e., considering d to be extremely heterogeneous), we get the standard passage language model from Equation 2 (page 5), which is the language model used by Liu and Croft [28]. On the other hand, assuming d is highly homogeneous and setting $h^{[\mathcal{M}]}(d) = 1$ results in representing each of d's passages with d's standard language model from Equation 2; note that in this case, the Max-Scoring Passage algorithm reduces to a standard document-based language model retrieval approach.

6 Passage-Based Document Representations

In this chapter we take a different view on the passage-based retrieval models from the previous chapters. We show, using our general passage-based retrieval framework from Section 4, that some of our passage-based document-ranking approaches are in fact standard document retrieval methods that utilize a certain form of document representation. While a document representation may be any composition of actual document parts or a synthetic derivation from it, we confine our discussion to passage-based document representations. In the following sections we describe how such representations can be utilized in ad-hoc document retrieval and propose a method for deriving passage-based document representations.

6.1 Representing documents

The great majority of classic IR systems were designed for use with bibliographic databases; indexing was applied to some document representation such as title, abstract or selected keywords, rather than to the document as a whole (a.k.a full text indexing), and document retrieval was based on matching a document representation with a query [17]. Steep decrease in storage costs and increase in available computation power over the years led to the development of full-text retrieval systems that allow full-text indexing and search. In this chapter we show that some of the algorithms presented in Chapter 4 can be formulated in terms of ad hoc document retrieval using *passage-based representation for documents*.

Document representation can be derived in a variety of ways. A representation may be based on one or more of the document's passages or sentences, as well as on a semantic entity such as the document title, written abstract or table of contents. It may also be the case that a representation is not an actual part of a document, but is synthesized from it by means of summary [29] or any other process.

Since our interest in this thesis lies in the realm of passage-based document retrieval we will confine our discussion to a specific type of document representation, which is based on the document's most representative passages. In the following formulation, we assume that each document is represented by a single passage r_d[6]. We show that using our new passage language model from Chapter 5 in the Max-Scoring Passage algorithm is equivalent to choosing this representative passage and performing standard

[6]This is akin to a summarization approach, where extracted key-phrases are used to create a document summary [29, 4].

document-based retrieval based on this representation. We discuss how a single representative passage r_d can be selected in Section 6.2.

We begin our discussion by referring the reader to Equation 12 (page 19), where we define a passage language model that incorporates information from the ambient document. If, for example, we use our Max-Scoring Passage algorithm and assign each passage a score according to the language model from Equation 12, then our retrieval approach is equivalent to deriving a document language model that is smoothed by the document highest-scoring passage model as formulated here

$$g^* = \operatorname*{argmax}_{g_j \in d} \prod_{q_i \in q} (\lambda_{psg}(g_j)\widetilde{p}^{MLE}_{g_j}(q_i) + \lambda_{doc}(d)\widetilde{p}^{MLE}_{d}(q_i) + \lambda_{\mathcal{C}}\widetilde{p}^{MLE}_{\mathcal{C}}(q_i));$$

$$p_d^{[g^*,\mathcal{M}]}(w) \stackrel{def}{=} \lambda_{psg}(g^*)\widetilde{p}^{MLE}_{g^*}(w) + \lambda_{doc}(d)\widetilde{p}^{MLE}_{d}(w) + \lambda_{\mathcal{C}}\widetilde{p}^{MLE}_{\mathcal{C}}(w)$$

The equation above implies that the Max-Scoring Passage algorithm can be viewed as a two-step retrieval process. At the first step we find the highest-scoring passage of the document, and at the second step we score the document based on a representation that depends on this passage.

If we were to design an analogous retrieval process wherein there is no dependency on the query at the first step, we could define our retrieval process in the following manner. First, we select a passage from the document according to some criteria (which will be discussed in the next section) and base our document representation on this passage. Then, we construct a language model based on this representation.

$$p_d^{[r_d,\mathcal{M}]}(w) \stackrel{def}{=} \lambda_{psg}(r_d)\widetilde{p}^{MLE}_{r_d}(w) + \lambda_{doc}(d)\widetilde{p}^{MLE}_{d}(w) + \lambda_{\mathcal{C}}\widetilde{p}^{MLE}_{\mathcal{C}}(w) \quad (13)$$

The principal difference between the Max-Scoring Passage algorithm and the retrieval process just described is that the former selects the passage that can potentially serve as a basis for a document representation based on a query (refer back to Section 4.1), while the latter assumes that only a single passage, r_d, can represent a document, i.e.

$$\hat{p}(g_i|d) \stackrel{def}{=} \begin{cases} 1 & \text{if } g_i \text{ is } r_d \\ 0 & \text{otherwise} \end{cases}$$

Since passage-based retrieval may significantly increase the cost of query evaluation [19], using query-independent document representations may ameliorate this problem. As opposed to finding the highest scoring passage of

each document for each query, representations are calculated only once and stored for use with all subsequent queries.

6.2 Passage selection for document representation

We will now discuss a method for selecting a passage r_d for deriving a query-independent passage-based document representation. Although the number of ways to derive a representation is virtually unlimited, we will focus on a single method, which mirrors one of the document homogeneity measures presented in Section 4.2.

A natural way to represent a document is by its summary or abstract, as indeed has been done in the classical information retrieval literature [4, 29]. As we are interested in introducing passages as means of representation, our aim is to find a document passage that would best fit the role of document summary or abstract.

Summary is usually defined as a process of reducing document complexity and length, while retaining some of the essential original qualities [20]. A good summary should facilitate quick and accurate identification of the original topic [29, 20]. Accordingly, we seek to represent an original (document) by a summary (passage) that bears the closest resemblance to it. To formally capture this notion we use a tf.idf vector-space representation [36] for both the document and all its passages and denote this represenation of text x as $\vec{x}$. We choose the passage g_i for which the cosine measure $\cos(\vec{d}, \vec{g_i})$ is maximized, to be the document representation r_d

$$r_d \stackrel{def}{=} \{g_i \in d : \ \forall j \ \cos(\vec{d}, \vec{g_i}) \geq \cos(\vec{d}, \vec{g_j})\}$$

This is reminiscent of $h^{[docPsg]}$ (see Section 4.2) — the document homogeneity measure where homogeneity is determined by the average cosine similarity of all the document's passages with the document as a whole. Indeed, the process of passage selection for document representation and document homogeneity are interconnected. Intuitively, the more homogeneous the document is, the higher is the probability that each of its passages may serve as a basis for a good document representation. Consider a highly homogeneous document, which is simply a concatenation of several copies of the same passage, as an example. In such a document, according to the definition of r_d above, each passage may serve as a document representation. On the other hand, for a heterogeneous document, where there is a large variance among passages' contents, a single passage that summarizes the ideas presented in other passages (such as an abstract), is a most likely candidate to become the basis for an effective document representation.

Once a document representation r_d has been selected, we can utilize it to perform *passage-representation based document retrieval*, i.e., we can assign a query-similarity score to a document based on a query-similarity score of its selected representation. Specifically, we choose to score a document in response to a query based on a unigram language model (see Chapter 2), as defined by Equation 13 (page 21)

$$\begin{aligned} r_d &\stackrel{def}{=} \{g_i \in d : \ \forall j \ \cos(\vec{d}, \vec{g_i}) \geq \cos(\vec{d}, \vec{g_j})\}; \\ p_d^{[r_d,\mathcal{M}]}(w) &\stackrel{def}{=} \lambda_{psg}(r_d)\widetilde{p}_{r_d}^{MLE}(w) + \lambda_{doc}(d)\widetilde{p}_d^{MLE}(w) + \lambda_{\mathcal{C}}\widetilde{p}_{\mathcal{C}}^{MLE}(w); \\ p_d^{[r_d,\mathcal{M}]}(w_1 w_2 \cdots w_n) &\stackrel{def}{=} \prod_{j=1}^{n} p_d^{[r_d,\mathcal{M}]}(w_j). \end{aligned}$$

We report the experimental results obtained by using this model in Chapter 7.

7 Evaluation

In what follows we present an experimental evaluation designed to explore the relative merits (or lack thereof) in using our proposed passage language model from Equation 12 (page 19) and the different passage-based ranking algorithms proposed in this thesis.

The rest of this chapter is organized as follows. In Section 7.1 we review the specific algorithm implementations we use in our experiments. In Section 7.2 we discuss the experimental setup. In Sections 7.3, 7.4, 7.5 and 7.6 we present detailed results for various algorithms. In Section 7.7 we summarize our experimental results and draw conclusions.

7.1 Algorithms overview

In this section we give a brief overview of the algorithms that were implemented for our experiments. All algorithm instantiations are based on the derivations made in Chapters 4, 5 and 6. All our algorithms are designed for document retrieval, and most of them rely to some extent on passage-based information. Whenever we combine document and passage statistics (as in Max-Scoring Passage, Interpolated Max-Scoring Passage or Representation-Based Scoring algorithms) for assigning a score to a document, we use our homogeneity measures (refer back to Section 4.2 for survey of measures used) for balancing the two.

We test the following algorithms for passage-based document retrieval:

- *Mean Passage-Scoring algorithm* — assigns a score to a document by the mean query-similarity score of its constituent passages. (See Equation 8, page 14.)
- *Max-Scoring Passage algorithm* — assigns a score to a document by the maximal query-similarity score of any of its passages. (See Equation 9, page 14.)
- *Interpolated Max-Scoring Passage algorithm* — assigns a score to a document by interpolating the query-similarity score of the document and the score derived from the Max-Scoring Passage algorithm. (See Equation 10, page 15.)
- *Representation-Based Scoring algorithm* — uses a single passage as a basis for document representation; assigns a score to a document by the query-similarity score of its passage-based representation. (See Equation 13, page 21.)

We instantiate each of the algorithms using document and passage language models (see Section 2 for details). Document language model is the basic language model $p_d^{[basic]}(\cdot)$ (Equation 1, page 5); passage language model is either the basic passage language model $p_g^{[basic]}(\cdot)$ (Equation 1, page 5) or our homogeneity-based passage language model $p_g^{[\mathcal{M}]}(\cdot)$ (Equation 11, page 19). The different algorithm instantiations are summarized in Figure 1 below.

Abbreviation	Language Model	Description
$BaseDoc$	$p_d^{[basic]}(\cdot)$	Standard language-model based document retrieval.
$MaxPsg$	$p_g^{[basic]}(\cdot)$	Max-Scoring Passage algorithm using a standard passage language model (Liu and Croft [28]).
$MeanPsg$	$p_g^{[basic]}(\cdot)$	Mean Passage-Scoring algorithm using standard passage language model.
$MSP[\mathcal{M}]$	$p_g^{[\mathcal{M}]}(\cdot)$	Max-Scoring Passage algorithm using our homogeneity-based passage language model.
$RelDoc$	$p_d^{[basic]}(\cdot)$	Standard document-based relevance model [25].
$RelPsg$	$p_g^{[basic]}(\cdot)$	Passage-based relevance model (Liu and Croft [28]).
$RelPsg[\mathcal{M}]$	$p_g^{[\mathcal{M}]}(\cdot)$	Passage-based relevance model, which utilizes our homogeneity-based language model.
$IMSP^{[\mathcal{M}]}[basic]$	$p_g^{[basic]}(\cdot)$, $p_d^{[basic]}(\cdot)$	Interpolated Max-Scoring Passage algorithm, wherein scores obtained by $BaseDoc$ and $MaxPsg$ are interpolated using homogeneity measures (see Equation 10).
$IMSP^{[\mathcal{M}]}[\mathcal{M}]$	$p_g^{[\mathcal{M}]}(\cdot)$, $p_d^{[basic]}(\cdot)$	Interpolated Max-Scoring Passage algorithm, wherein scores obtained by $BaseDoc$ and $MSP[\mathcal{M}]$ are interpolated using homogeneity measures. (See Equation 10).
$Rep[\mathcal{M}]$	$p_g^{[\mathcal{M}]}(\cdot)$	Representation-Based Scoring algorithm, wherein score of a document is determined by the score of its representation, as in Equation 13, page 21.

Figure 1: **Summary of all the evaluated algorithm instantiations. Each instantiation is represented by an abbreviated name. For each instantiation the following information is provided: language model(s) it uses and a brief textual description.**

7.2 Experimental setup

Corpora We conducted our experiments on the following four TREC corpora (we considered only queries in the specified ranges with at least one relevant document):

corpus	# of docs	avg. doc. length	queries	disk(s)
FR12	45,820	935	51-100	1,2
LA+FR45	187,526	317	401-450	4,5
WSJ	173,252	263	151-200	1-2
AP89	84,678	264	1-50	1

We note that FR12, which was used in work on passage-based document retrieval [7, 28], and LA+FR45, which is known to be a very hard benchmark with TREC8 queries (401-450) [14, 23], are considered to contain highly heterogeneous documents, while documents in AP89 and WSJ are considered to be more homogeneous.

We used the Lemur[7] toolkit to run our experiments. We applied basic tokenization and Porter stemming, and removed INQUERY stopwords [3]. We used the titles of TREC topics as queries.

Evaluation metrics To evaluate retrieval performance, we use the following metrics: mean average (non-interpolated) precision at 1000 (MAP), precision at 5 documents (p@5) and precision at 10 documents (p@10). Mean average precision is a widely accepted metric for the evaluation of the general quality of retrieval methods [43]; p@5 and p@10 metrics measure the ability of retrieval methods to position relevant documents at the very high ranks of the retrieved results. We determine statistically significant differences in performance using the two-tailed Wilcoxon test at the 95% confidence level.

Passages While there are several passage types we can choose from (refer back to Section 3.1.1), our focus here is on the general validity of our retrieval algorithms and language-model induction techniques. Therefore, we use *half overlapping fixed-length windows* of sizes 150, 50 and 25 as passages and mark them *prior* to retrieval time. Such passages are computationally convenient to use and were shown to be quite effective for document retrieval [7], specifically in the language model framework when compared to other passage types [28].

[7] www.lemurproject.org

Baselines In most of our experiments we use the following reference comparisons for our algorithms. The first is a standard language-model-based document retrieval, denoted $BaseDoc$, wherein d is scored by $p_d^{[basic]}(q)$. The second is the Max-Scoring Passage algorithm, implemented using the standard passage language model $p_g^{[basic]}(q)$, which we denote as $MaxPsg$. (The latter was proposed by Liu and Croft [28]). Note that both references are applications of the query-likelihood approach, either for documents ($BaseDoc$) or for passages ($MaxPsg$) (see Section 2.1 for details).

Parameter tuning All tested algorithm implementations use a single free parameter $\lambda_{\mathcal{C}}$, which controls the amount of reliance on corpus-based statistics for smoothing.

To establish a fair comparison of our algorithms' implementations with the basic (unsmoothed) passage and document language models they utilize, we choose the value of $\lambda_{\mathcal{C}}$ from $\{0.1, \ldots, 0.9\}$ for which the MAP performance of *both* $MaxPsg$ and $BaseDoc$ (our reference comparisons) is nearly optimal; the performance results for using different values of $\lambda_{\mathcal{C}}$ for these two algoritms on all tested corpora are presented in Figure 2. (Passage size is 150. Similar trends exist for passage sizes 50 and 25, however they were not plotted to avoid cluttering the graphs.) As can be seen in Figure 2, results for $\lambda_{\mathcal{C}} = 0.5$ are near optimal in most cases both for $BaseDoc$ and $MaxPsg$; hence we will set the value of $\lambda_{\mathcal{C}}$ to 0.5 in all succeeding experiments.

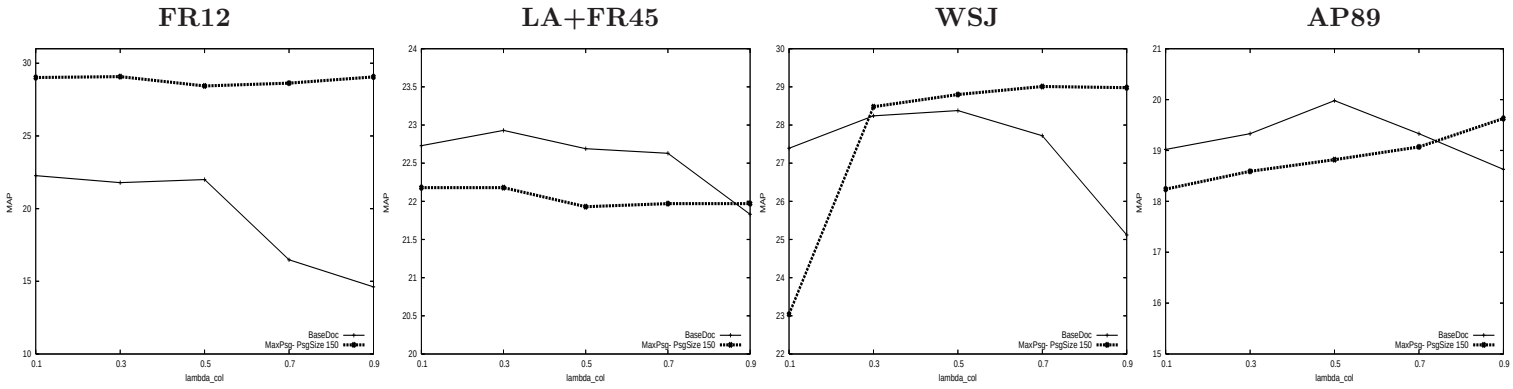

Figure 2: **MAP performance numbers of $BaseDoc$ and $MaxPsg$.** $BaseDoc$ **and** $MaxPsg$ **are represented by thin and thick lines respectively. Performance is shown for passage size 150 when setting $\lambda_{\mathcal{C}}$ to $\{0.1, \ldots, 0.9\}$. Note: figures are not to the same scale.**

As a note aside, we must mention that while setting $\lambda_{\mathcal{C}}$ to a fixed value re-

sults in a fair comparison between the algorithms, it also results in $MaxPsg$ and $BaseDoc$ utilizing Jelinek-Mercer smoothing (refer back to Chapter 2), which is somewhat less effective than Dirichlet smoothing when using short queries [46]. However, having our algorithms incorporate the length-based Dirichlet smoothing effect [46], calls for a somewhat different formulation due to issues related to the normalization of the $h^{[\mathcal{M}]}$ function and the statistical interpretation of the prior. We compare, however, the $MaxPsg$ and $BaseDoc$ algorithms with Dirichlet smoothing to our Interpolated Max-Scoring Passage algorithm in Section 7.5.

7.3 The Mean Passage-Scoring algorithm

In our first set of experiments we consider the Mean Passage-Scoring algorithm (see Equation 6, page 14), wherein we use a weighted sum of query-similarity passage scores to rank a document. In lack of any information regarding the relative importance of passages we assume they have an equal importance for scoring purposes, and score a document by the mean query-similarity score of its constituent passages as in Equation 8 (page 14); thus, we get the Mean Passage-Scoring algorithm (refer to Chapter 5).

In our experiments we use the standard language model $p_g^{[basic]}(\cdot)$ (see Equation 1, page 5). This means that for creating a language model we treat each passage as a separate piece of text, and smooth it's statistics only with the collection statistics. We use Jelinek-Mercer smoothing [46] and set $\lambda_{\mathcal{C}} = 0.5$, as described in Section 7.2; we compare the resultant performance to that of $BaseDoc$ and $MaxPsg$ — our reference comparisons (see Section 7.2).

The performance numbers in Figure 3 clearly indicate the inferiority of the proposed Mean Passage-Scoring algorithm to using both standard document language model and the Max-Scoring Passage algorithm with standard passage language model; the performance numbers for all evaluation metrics are lower than those of the reference comparisons for all corpora (up to two times and more — see the results for FR12 and WSJ).

These results resonate with the hypothesis (see Section 4.1) that retrieval performance can be enhanced by a detection of a *single* relevant passage per document rather than by an utilization of all document passages. This is further reinforced by the fact that Max-Scoring Passage algorithm ($MaxPsg$) outperforms the document retrieval ($BaseDoc$) on some corpora that are considered heterogeneous (see the results for FR12 corpus, for example). Accordingly, we next move to the evaluation of our implementation of the Max-Scoring Passage algorithm, which uses our homogeneity-based passage language model.

FR12

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	22.00	**19.05**	13.33	22.00	**19.05**	13.33	**22.00**	**19.05**	13.33
MaxPsg	**28.44**	**19.05**	**14.76**	**30.14** ^{d}	**19.05**	**14.76**	18.10	13.33	**13.81**
MeanPsg	13.89^{d}_{p}	10.48^{d}_{p}	9.52_{p}	13.38^{d}_{p}	7.62^{d}_{p}	6.67^{d}_{p}	9.31^{d}_{p}	5.71^{d}	6.19^{d}_{p}

LA+FR45

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	**22.69**	**30.64**	**26.38**	**22.69**	**30.64**	**26.38**	**22.69**	**30.64**	26.38
MaxPsg	21.93	27.66	25.53	21.68	28.51	25.74	21.71	29.36	**26.81**
MeanPsg	20.22^{d}_{p}	25.96^{d}	22.98^{d}	16.26^{d}_{p}	19.15^{d}_{p}	18.09^{d}_{p}	14.64^{d}_{p}	17.87^{d}_{p}	17.23^{d}_{p}

WSJ

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	28.38	42.40	39.60	**28.38**	42.40	39.60	**28.38**	**42.40**	**39.60**
MaxPsg	**28.80**	**46.00**	**41.80**	26.10^{d}	**44.00**	**40.40**	24.95^{d}	40.80	37.20
MeanPsg	20.42^{d}_{p}	32.00^{d}_{p}	32.60^{d}_{p}	14.03^{d}_{p}	21.20^{d}_{p}	19.60^{d}_{p}	10.72^{d}_{p}	14.40^{d}_{p}	14.20^{d}_{p}

AP89

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	**19.98**	25.65	**24.13**	**19.98**	25.65	**24.13**	**19.98**	**25.65**	**24.13**
MaxPsg	18.82^{d}	**27.83**	23.04	17.71^{d}	**26.09**	22.39	16.34^{d}	20.43	16.52^{d}
MeanPsg	17.55^{d}_{p}	21.74_{p}	22.39	14.27^{d}_{p}	14.78^{d}_{p}	13.48^{d}_{p}	11.99^{d}_{p}	12.17^{d}_{p}	10.87^{d}_{p}

Figure 3: **Performance numbers of the Mean Passage-Scoring algorithm ($MeanPsg$).**
Performance numbers for retrieval using basic document language model ($BaseDoc$) and the Max-Scoring Passage algorithm with basic passage language model ($MaxPsg$), are presented for reference. Boldface indicates the best result per column; shadow marks the best performance in a table with respect to an evaluation measure; d and p mark statistically significant differences with $BaseDoc$ and $MaxPsg$ respectively.

7.4 Our passage language model

In this section we explore the benefits (or lack thereof) in using our homogeneity-based passage language model from Equation 12 (page 19), $p_g^{[\mathcal{M}]}(\cdot)$ in the Max-Scoring Passage algorithm and in constructing and utilizing passage-based relevance models [28].

7.4.1 The Max-Scoring Passage algorithm

We use $MSP[\mathcal{M}]$ to denote the implementation of Max-Scoring Passage with our homogeneity-based language model, and compare its performance to that obtained by our reference comparisons $BaseDoc$ and $MaxPsg$.

Recall our observation from Chapter 5 that fixing $h^{[\mathcal{M}]}(d)$ to either 0 or 1 in Equation 11 (page 19) (i.e., assuming d is either highly heterogeneous or highly homogeneous) and using the Max-Scoring Passage algorithm with the resultant passage language model amounts to ranking by $MaxPsg$ and $BaseDoc$ respectively. Thus, we get that our two reference comparisons are in fact specific instantiations of Max-Scoring Passage with degenerated homogeneity measures.

We now turn to Figure 4, in which we present the performance numbers for the above mentioned methods.

Our first observation in Figure 4 is that the Max-Scoring Passage algorithm is consistently more effective when utilizing our new passage language model $p_g^{[\mathcal{M}]}(\cdot)$ than when using the standard passage language model $p_g^{[basic]}(\cdot)$. We can observe the following for the 48 relevant comparisons (4 corpora $\times$ 4 homogeneity measures $\times$ 3 passage sizes)

- For the MAP evaluation metric, $MSP[\mathcal{M}]$ is superior to $MaxPsg$ in about 96% of the cases
- For the p@5 evaluation metric, $MSP[\mathcal{M}]$ is superior (or equal) to $MaxPsg$ in about 65% of the cases
- For the p@10 evaluation metric, $MSP[\mathcal{M}]$ is superior to $MaxPsg$ in about 80% of the cases

In many cases, the performance differences are also statistically significant. (See the AP89 and the WSJ cases, for example.)

Another observation we make based on Figure 4 is that the best performing document homogeneity measures for inducing our passage model are *length* — demonstrating its correlation with heterogeneity [39] — and

docPsg, which measures the similarity between a document and its passages, and is thus directly related to the balance of document-based and passage-based information that we want to control in inducing the passage language model.

This observation on document homogeneity measures performance is further strengthened by the following significance tests between the MAP performance results obtained using different homogeneity measures: (i) both $MSP[length]$ and $MSP[docPsg]$ performances are better to a statistically significant degree than that of $MSP[interPsg]$ for FR12 (passage sizes 150 and 50), (ii) $MSP[length]$ performance is better to a statistically significant degree than that of $MSP[interPsg]$ and $MSP[ent]$, and $MSP[docPsg]$ performance is better to a statistically significant degree than that of $MSP[ent]$ for LA+FR45 (passage size 50), (iii) both $MSP[length]$ and $MSP[docPsg]$ performances are better to a statistically significant degree than that of $MSP[interPsg]$ and $MSP[ent]$ for WSJ (passage size 50), and (iv) $MSP[length]$ performance is better to a statistically significant degree than that of $MSP[ent]$, and $MSP[docPsg]$ performance is better to a statistically significant degree than that of $MSP[interPsg]$ and $MSP[ent]$ for AP89 (passage size 150).

We can also see in Figure 4 that very short passages (of size 25) are the worst choice among the three we consider, which is in line with some previous results [7, 28]. However, using our passage language model that incorporates document statistics ameliorates the performance decay for small passage size to some degree when compared to standard passage language model. (This is further analyzed in Section 7.4.2). Among the tested passage sizes, passages of size 50 provide the optimal performance.

Our best performing methods, $MSP[length]$ and $MSP[docPsg]$, are both superior in 75% of the relevant comparisons (4 corpora $\times$ 3 evaluation metrics) to the standard document-based method, $BaseDoc$, for passage size 50; sometimes (e.g., on FR12), the differences are also statistically significant.

Therefore, perhaps the most important conclusion we can draw from Figure 4 is that using our passage language model, which integrates passage and document information, in the Max-Scoring Passage algorithm results in performance that is in many times better than that resulting from the use of either the standard passage language model or the standard document language model. (The performance numbers in the FR12 and LA+FR45 tables nicely illustrate this conclusion: $MSP[length]$ is superior in most of the relevant comparisons to both $MaxPsg$ and $BaseDoc$, while $MaxPsg$ is superior to $BaseDoc$ on FR12, and inferior to $BaseDoc$ on LA+FR45.)

FR12

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	22.00	**19.05**	13.33	22.00	19.05	13.33	22.00	19.05	13.33
MaxPsg	28.44	**19.05**	14.76	30.14^{d}	19.05	14.76	18.10	13.33	13.81
MSP[length]	$\mathbf{29.56}^{d}$	18.10	15.71	$\mathbf{31.83}_{p}^{d}$	**20.00**	15.71	$\mathbf{26.87}_{p}$	$\mathbf{21.90}_{p}$	16.19
MSP[ent]	29.25^{d}	**19.05**	**16.19**	30.12^{d}	18.10	**16.19**	25.08	20.00	**16.67**
MSP[docPsg]	29.32^{d}	**19.05**	**16.19**	31.01^{d}	19.05	15.71	25.32	18.10	12.86
MSP[interPsg]	29.05^{d}	18.10	15.71	30.70^{d}	18.10	**16.19**	25.37	18.10	12.38

LA+FR45

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	22.69	**30.64**	26.38	22.69	**30.64**	**26.38**	22.69	**30.64**	26.38
MaxPsg	21.93	27.66	25.53	21.68	28.51	25.74	21.71	29.36	**26.81**
MSP[length]	23.05_{p}	28.94	27.45	$\mathbf{23.56}_{p}$	28.94	25.96	**23.21**	25.53	25.11
MSP[ent]	22.20	27.66	26.17	21.83	29.79	25.96	21.87	28.94	25.74
MSP[docPsg]	$\mathbf{23.16}_{p}$	29.36	**27.87**	22.99	26.38	25.53	21.75	26.38	24.26
MSP[interPsg]	22.75_{p}	27.66	26.60	21.92	26.81	25.32	21.04	28.09	24.04_{p}

WSJ

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	28.38	42.40	39.60	28.38	42.40	39.60	**28.38**	42.40	39.60
MaxPsg	28.80	**46.00**	41.80	26.10^{d}	44.00	40.40	24.95^{d}	40.80	37.20
MSP[length]	29.25^{d}	44.40	$\mathbf{43.00}^{d}$	29.00_{p}	**46.00**	$\mathbf{44.80}_{p}^{d}$	27.91_{p}	**44.00**	$\mathbf{43.40}_{p}$
MSP[ent]	$\mathbf{29.32}_{p}$	44.00	41.60	27.86_{p}	**46.00**	41.80	26.49_{p}^{d}	41.60	39.60
MSP[docPsg]	29.13^{d}	44.40	42.60^{d}	$\mathbf{29.15}_{p}$	45.60	$\mathbf{44.80}_{p}^{d}$	27.80_{p}	42.00	42.40_{p}
MSP[interPsg]	29.20^{d}	45.20	42.40^{d}	28.16_{p}	45.20	43.20_{p}	26.83_{p}^{d}	42.00	38.40

AP89

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	**19.98**	25.65	**24.13**	**19.98**	25.65	24.13	**19.98**	**25.65**	**24.13**
MaxPsg	18.82^{d}	**27.83**	23.04	17.71^{d}	26.09	22.39	16.34^{d}	20.43	16.52^{d}
MSP[length]	19.32_{p}	**27.83**	23.70	18.74_{p}	26.09	**24.57**	17.79_{p}^{d}	24.78_{p}	20.87_{p}
MSP[ent]	19.05_{p}	**27.83**	22.83	18.24_{p}^{d}	26.09	22.61	17.35_{p}^{d}	20.87	19.57_{p}^{d}
MSP[docPsg]	19.75_{p}	26.09	23.26	19.06_{p}	**29.13**	**24.57**	17.73_{p}^{d}	24.78_{p}	21.96_{p}
MSP[interPsg]	19.47_{p}	25.65	23.70	18.39_{p}^{d}	24.78	23.91	17.42_{p}^{d}	22.61	19.35_{p}^{d}

Figure 4: **Performance numbers of the Max-Scoring Passage algorithm ($MSP[\mathcal{M}]$).**

Max-Scoring Passage algorithm is implemented either with the basic passage language model, $p_g^{[basic]}(\cdot)$, from Equation 2 (as in Liu and Croft [28]) — denoted $MaxPsg$, or with our passage language model, $p_g^{[\mathcal{M}]}(\cdot)$, with homogeneity model $\mathcal{M}$ — denoted $MSP[\mathcal{M}]$. Document-based language-model retrieval performance is presented for reference ($BaseDoc$). Boldface indicates the best result per column; shadow marks the best performance in a table with respect to an evaluation measure; d and p mark statistically significant differences with $BaseDoc$ and $MaxPsg$ respectively.

7.4.2 Further analysis

Effect of passage size on retrieval performance It is clear from Figure 4 that the selection of passage size has a considerable effect on retrieval performance. Similar effects were also reported in previous work on passage retrieval [28, 7, 18]. In general, we note that performance obtained for passages of size 50 is nearly optimal (with respect to other passage sizes we have tested) in almost all of the cases when using our passage language model (see, for example, the results for FR12 and WSJ). The retrieval performance is at its worst both for our passage language model and for the basic passage language model when passages of size 25 are used. However, as mentioned above, performance decay tends to be ameliorated to some degree when our passage language model, $p_g^{[\mathcal{M}]}(\cdot)$, is used. The results for FR12 and WSJ nicely illustrate this tendency. For FR12, when the basic passage language model is used, the decline between the best (for passage size 50) and the worst (for passage size 25) MAP performance is about 40%; by comparison, when our passage language model is used, the decline between the best and the worst MAP performance (as in the case for $MSP[length]$) is only about 15%. For WSJ, the decline between the best and the worst MAP performance is 15% and 5%, when using either standard language model, $p_g^{[basic]}(\cdot)$, or our language model, $p_g^{[\mathcal{M}]}(\cdot)$, respectively.

This ameliorating effect could stem from the incorporation of document statistics into our passage model, which restricts to some degree the performance deviation caused by varying the passage sizes. It must be noted, however, that since only three different passage sizes are used, there is not enough data to make a conclusive judgment on the exact role that document statistics might play in the selection of passage size.

Effect of homogeneity measures on retrieval performance We derived the passage language model in Equation 11 (page 19) by controlling the reliance on document versus passage information using homogeneity measure $\mathcal{M}$. We now examine the alternative of fixing the balance between the two, making the assumption that all documents in the corpus are homogeneous to the same extent; specifically we set $h^{[\mathcal{M}]}(d)$ to a fixed value in $\{0, 0.2, \ldots, 1\}$ for all $d \in \mathcal{C}$. Note that doing so results in fixing $\lambda_{doc}(d)$ in Equation 11 (page 19) to a value in $\{0, 0.1, \ldots, 0.5\}$ (since $\lambda_{doc}(d) = (1 - \lambda_{\mathcal{C}})h^{[\mathcal{M}]}(d)$ and $\lambda_{\mathcal{C}} = 0.5$), which echoes some past work [1, 15, 32, 44]. Furthermore, observe that setting $\lambda_{doc}(d)$ to 0 or 0.5 and using the Max-Scoring Passage algorithm amounts to using the reference comparisons $MaxPsg$ and $BaseDoc$ respectively. (Refer to Chapter 5.)

Figure 5 depicts the MAP performance curve of the Max-Scoring Passage algorithm (with passage sizes set to 150, 50 and 25) when varying $\lambda_{doc}(d)$. We also plot for comparison the performance of our best performing methods $MSP[docPsg]$ and $MSP[length]$, with thick and thin horizontal lines respectively.

We can see in Figure 5 that using the homogeneity measures with passage sizes 150 and 50 helps us to avoid a relatively poor performance obtained by a bad choice of a constant $\lambda_{doc}(d)$ as is the case for AP89. (Note that for FR12 and WSJ the worst choice amounts to using $BaseDoc$, while for LA+FR45 and AP89 it amounts to using $MaxPsg$.) Furthermore, for the other three corpora, using our homogeneity measure results in near (or even better than) optimal performance with respect to a fixed $\lambda_{doc}(d)$. It is also important to note that while the performance differences are small in absolute terms, many of them are statistically significant.

- For passage size 150, $MSP[length]$'s performance is better to a statistically significant degree than that resulting from setting $\lambda_{doc}(d) = 0$ for LA+FR45 and AP89, and $\lambda_{doc}(d) = 0.5$ for FR12 and WSJ. $MSP[docPsg]$ is better to a statistically significant degree than using $\lambda_{doc}(d) \in \{0.1, 0.2\}$ for AP89, and to $\lambda_{doc}(d) = 0.1$ for LA+FR45.

- For passage size 50, $MSP[length]$'s performance is better to a statistically significant degree than that resulting from setting: (i) $\lambda_{doc}(d) = 0$ for all corpora, (ii) $\lambda_{doc}(d) = 0.5$ for FR12, (iii) $\lambda_{doc}(d) = 0.1$ for AP89, FR12 and WSJ, and (iv) $\lambda_{doc}(d) = 0.1$ for FR12. $MSP[docPsg]$'s performance is better to a statistically significant degree than that resulting from setting $\lambda_{doc}(d) = 0$ for WSJ and AP89, $\lambda_{doc}(d) = 0.5$ for FR12, and $\lambda_{doc}d = 0.1$ for AP89 and WSJ.

We can also see in Figure 5 that when small passage size (25) is selected, values of $\lambda_{doc}(d)$ that provide the best performance tend to be closer to 0.5 — i.e., assign higher weight to document statistics in the language model in Equation 11 (page 19). (Note that setting $\lambda_{doc}(d) = 0.5$ results in $BaseDoc$ algorithm). This effect can be observed when comparing plots for different passage sizes in Figure 5. In plots depicting performance for FR12, LA+FR45 and WSJ for passage size 25, best performance is attained by setting $h^{[\mathcal{M}]}(d) = 0.4$; on the other hand in plots depicting performance for FR12, LA+FR45 and WSJ for passage size 150, best performance is attained by setting $h^{[\mathcal{M}]}(d) = 0.2$, $h^{[\mathcal{M}]}(d) = 0.3$ and $h^{[\mathcal{M}]}(d) = 0.2$ respectively. Due to the fact that the optimal values of $\lambda_{doc}(d)$ are close to 0.5 in the case of passages of size 25, homogeneity models performance is not as

good as for larger passage sizes, however they still sometimes provide near optimal performance with respect to fixed $\lambda_{doc}(d)$ (as in FR12), or at least help to avoid a degraded performance caused by selecting a low fixed value for $\lambda_{doc}(d)$(as in LA+FR45).

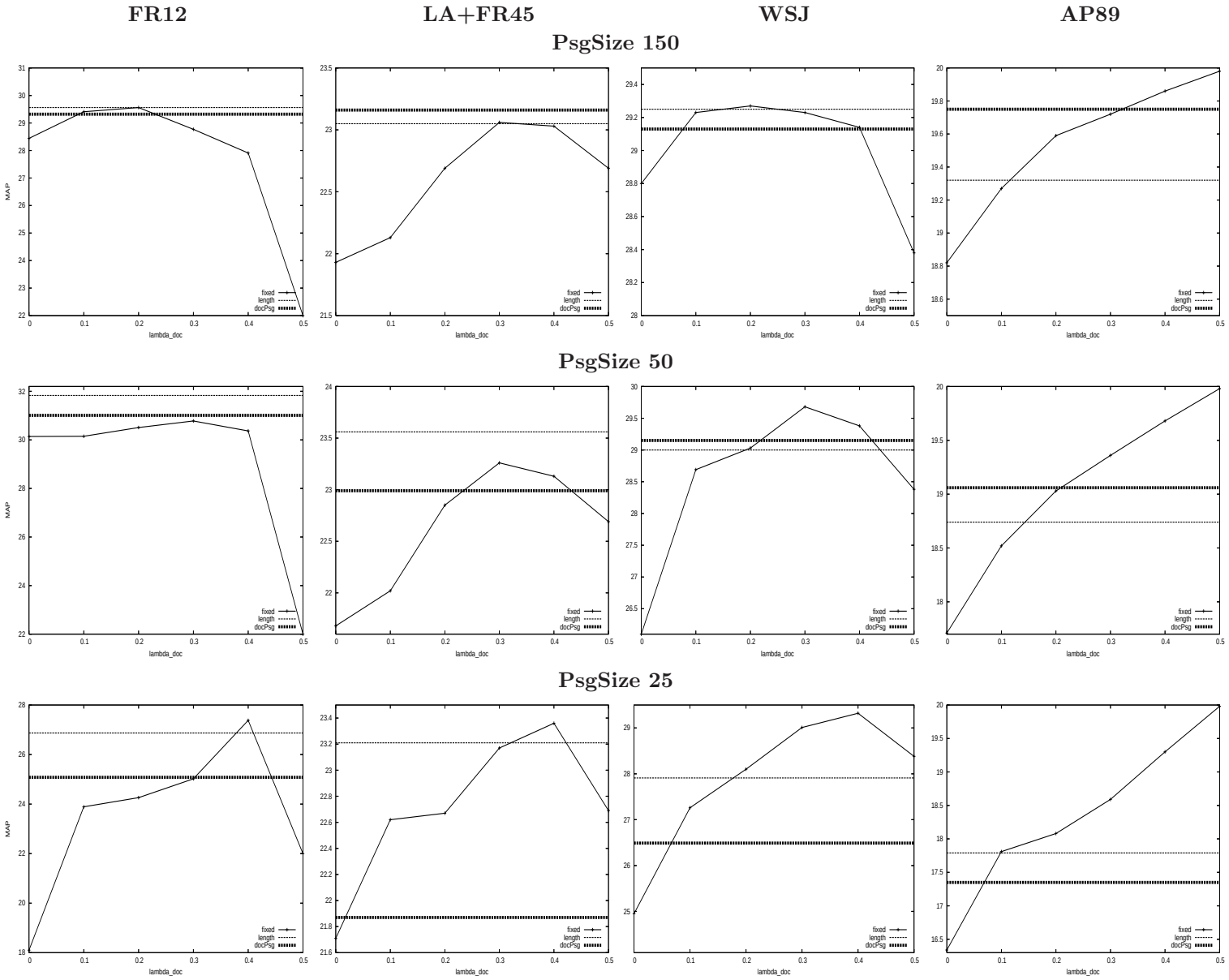

Figure 5: **The Max-Scoring Passage algorithm's MAP performance when either setting $h^{[\mathcal{M}]}(d)$ to fixed values or using homogeneity measures. The performance is shown when either setting $\lambda_{doc}(d)$ (from Equation 11) to a value in $\{0, 0.1, \ldots, 0.5\}$ for all $d \in \mathcal{C}$ (note that 0 and 0.5 correspond to $MaxPsg$ and $BaseDoc$ respectively), or using the homogeneity measures $length$ (thin horizontal line) and $docPsg$ (thick horizontal line) instead, although these measures do *not* incorporate free parameters. (Lines are used for convenience of comparison.) Note: figures are not to the same scale.**

7.4.3 Passage-based relevance models

Liu and Croft [28] suggest several methods for constructing and utilizing *passage-based relevance models* [25] for document retrieval. Their most effective method is to construct a relevance model $\mathcal{R}$ (see Section 2.2) using only passages, and then score each document $d \in \mathcal{C}$ by $min_{g_i \in d} D\left(\widetilde{p}_{\mathcal{R}}(\cdot) \,\middle|\middle|\, \widetilde{p}_{g_i}^{[basic]}(\cdot)\right)$ (see Equation 13, page 21). Conceptually, this algorithm is a special case of the Max-Scoring Passage algorithm, wherein q is replaced with $\mathcal{R}$.

We now compare their implementation, denoted $RelPsg$, which utilizes the standard passage language model $p_g^{[basic]}(\cdot)$, to an implementation, denoted $RelPsg[\mathcal{M}]$, which utilizes our new passage language model $p_g^{[\mathcal{M}]}(\cdot)$. We also use the standard document-based relevance model [25], denoted $RelDoc$, as a reference comparison.

We optimize the performance of each of our reference comparisons ($RelPsg$ and $RelDoc$) with respect to the number of top-retrieved elements (i.e., passages or documents) and the number of terms used for constructing the relevance models; specifically, we select these parameters' values from the set $\{25, 50, 75, 100, 250, 500\}$ – i.e., total of 36 parameters settings — so as to optimize MAP performance.

We set $\lambda_{\mathcal{C}} = 0.5$ (as in Section 7.2) except for estimating top-retrieved elements' language models for constructing $\mathcal{R}$, wherein we set $\lambda_{\mathcal{C}} = 0.2$ following previous observations [25]. To set parameters values for our $RelPsg[\mathcal{M}]$ algorithms we use those chosen for the $RelPsg$ reference comparison. Thus, our $RelPsg[\mathcal{M}]$ algorithms performance is not necessarily the optimal one they can potentially achieve. We also point out that the performance of the document-based relevance model $RelDoc$ might be further improved by employing Dirichlet smoothing rather than Jelinek-Mercer smoothing; however, having all tested language models employ the same smoothing technique is crucial for studying their relative effectiveness. (Refer back to the discussion at the end of Section 7.2.)

We present the performance results for the different relevance models in Figure 6. We see in Figure 6 that in a vast majority of the relevant comparisons, using our passage language model results in relevance models ($RelPsg[\mathcal{M}]$) that outperform both the one utilizing the previously suggested basic passage model ($RelPsg$), and the document-based relevance model ($RelDoc$). (Observe, for example, that shadows that mark the best performance in a table per evaluation metric appear almost exclusively in $RelPsg[\mathcal{M}]$ rows.) Furthermore, in many of the comparisons, the performance differences are also statistically significant, especially for passages of

size 50. Among the outstanding performance results for passage size 50, are the results attained by $RelPsg[docPsg]$ for FR12 and WSJ, where it outperforms (in terms of MAP) the best-performing of the two baselines, $RelDoc$ and $RelPsg$, by 9.3% and 9.7%, respectively.

FR12

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
RelDoc	10.70	10.48	9.05	10.70	10.48	9.05	10.70	10.48	9.05
RelPsg	**31.71**^{d}	17.14	14.29^{d}	31.06^{d}	19.05	16.19^{d}	22.42^{d}	13.33	14.29
RelPsg[length]	28.00^{d}	19.05	14.76^{d}	30.74^{d}	**23.81**^{d}	**18.10**^{d}	28.89^{d}	**21.90**$^{d}_{p}$	**17.62**^{d}
RelPsg[ent]	29.36^{d}	**20.00**^{d}	14.76^{d}	33.41^{d}	22.86^{d}	**18.10**^{d}	**31.55**$^{d}_{p}$	20.00	17.14^{d}
RelPsg[docPsg]	26.86^{d}	18.10	**15.71**^{d}	**34.23**^{d}	22.86^{d}	**18.10**^{d}	28.44^{d}	20.95	17.14^{d}
RelPsg[interPsg]	29.32^{d}	19.05^{d}	14.76^{d}	33.08^{d}	**23.81**^{d}	17.62^{d}	22.47^{d}	16.19	15.24^{d}

LA+FR45

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
RelDoc	20.69	28.09	23.83	20.69	28.09	23.83	20.69	28.09	23.83
RelPsg	**22.44**	28.94	25.96	21.87	29.79	24.68	20.26	28.09	22.98
RelPsg[length]	21.77$_{p}$	**31.91**	**26.60**	**23.30**$^{d}_{p}$	33.19	25.32	22.02$^{d}_{p}$	**30.64**	24.47
RelPsg[ent]	22.30	29.79	**26.60**	23.05	32.77	25.53	**22.23**$_{p}$	30.21	24.26
RelPsg[docPsg]	20.35$_{p}$	30.64	25.11	22.79$^{d}_{p}$	**34.04**$_{p}$	**25.74**	21.40$^{d}_{p}$	29.36	**24.68**
RelPsg[interPsg]	21.62	30.21	25.96	23.01	33.19$_{p}$	25.32	22.08	**30.64**	**24.68**

WSJ

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
RelDoc	33.85	48.80	48.40	33.85	48.80	48.40	33.85	48.80	**48.40**
RelPsg	34.46	50.40	47.20	33.97	47.20	45.00	30.95	43.60	41.60^{d}
RelPsg[length]	35.40^{d}	**54.40**$_{p}$	50.00	37.53$^{d}_{p}$	49.60	49.00$_{p}$	35.63$_{p}$	49.20	47.20$_{p}$
RelPsg[ent]	34.90^{d}	52.00	48.20	36.86$^{d}_{p}$	49.60	46.80	34.10$_{p}$	48.80	46.40$_{p}$
RelPsg[docPsg]	**35.90**^{d}	52.40	**50.20**	**37.60**$^{d}_{p}$	49.60	**50.20**$_{p}$	**35.89**$_{p}$	48.00	47.80$_{p}$
RelPsg[interPsg]	35.25^{d}	52.40	50.00	37.47$^{d}_{p}$	**50.00**$_{p}$	47.20	34.66$_{p}$	**49.60**$_{p}$	45.00$_{p}$

AP89

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
RelDoc	25.56	31.30	28.48	**25.56**	31.30	28.48	**25.56**	31.30	**28.48**
RelPsg	24.08	**35.65**	29.78	22.18	30.43	25.87	20.45^{d}	25.65	22.83^{d}
RelPsg[length]	25.05	33.48	**30.43**	24.30$_{p}$	33.48	30.00$_{p}$	22.33$^{d}_{p}$	30.87$_{p}$	27.61$_{p}$
RelPsg[ent]	24.54	34.35	30.00	23.68$_{p}$	**35.22**$_{p}$	29.35	21.67$^{d}_{p}$	31.30	26.09$_{p}$
RelPsg[docPsg]	**25.73**	32.61	29.78	25.13$_{p}$	34.78	**31.30**$_{p}$	23.04$^{d}_{p}$	**32.61**$_{p}$	28.26$_{p}$
RelPsg[interPsg]	25.24	31.74	**30.43**	24.10$_{p}$	33.48	29.13	22.07$^{d}_{p}$	31.30$_{p}$	26.52$_{p}$

Figure 6: **Performance numbers of *passage-based relevance model*. Either the originally suggested by Liu and Croft [28] basic passage language model, $p_g^{[basic]}(\cdot)$, ($RelPsg$) or our new passage language model, $p_g^{[\mathcal{M}]}(\cdot)$, ($RelPsg[\mathcal{M}]$) is used for *passage-based relevance model* instantiation. Document-based relevance-model performance is presented for reference ($RelDoc$). Best result in a column is boldfaced, and best result in a table (per evaluation measure) is marked with a shadow; significant differences with $RelDoc$ and $RelPsg$ are marked with d and p respectively.**

7.4.4 Conclusions

In this section we used our homogeneity-based language model $p_g^{[\mathcal{M}]}(\cdot)$ to instantiate the Max-Scoring Passage algorithm ($MSP[\mathcal{M}]$). Using our language model in Max-Scoring Passage algorithm resulted in (for many cases statistically significant) performance improvements over our reference comparisons $BaseDoc$ and $MaxPsg$. We also compared the effectiveness of setting $h^{[\mathcal{M}]}(d)$ for all the documents to a fixed value (as is done in some previous work [1, 15, 32, 44]) with that of setting $h^{[\mathcal{M}]}(d)$ according to document homogeneity (as in our homogeneity-based passage language model). We showed that for larger passage sizes, using our homogeneity-based passage language model results in near (or even better than) optimal performance with respect to any fixed weight.

In addition, we used our language model $p_g^{[\mathcal{M}]}(\cdot)$ to construct a new *relevance model* [25]. Document retrieval using our new relevance model ($RelPsg[\mathcal{M}]$) demonstrated (in many cases statistically significant) performance improvement over retrieval using both a document-based *relevance model* [25] and a passage-based *relevance model* constructed using a standard passage language model $p_g^{[basic]}(\cdot)$ [28] (see Figure 6).

7.5 The Interpolated Max-Scoring Passage algorithm

The Interpolated Max-Scoring Passage algorithm scores document d by the interpolation of the standard document-based language model score ($BaseDoc$) with the score derived from the Max-Scoring Passage algorithm (refer to Section 5). We experimented with two versions of the Interpolated Max-Scoring Passage algorithm.

In the first version, which focuses solely on the score integration, the standard document and passage language models from Equation 2 (page 5) are used. Thus, the Max-Scoring Passage implementation in this case is Liu and Croft's algorithm [28] — denoted $MaxPsg$ in Section 7.2. The second version interpolates the standard document language model score ($BaseDoc$) and the score assigned by the Max-Scoring Passage algorithm when implemented using our passage language model (see Equation 11, page 19).

Thus, while the first version of the algorithm is designed to test whether performances of standard document or passage language models ($BaseDoc$ and $MaxPsg$, respectively) can be improved by means of score interpolation, the second version explores whether score interpolation might bring further improvements over performance gains already obtained by using the Max-Scoring Passage algorithm with the new passage language model (refer to the results for $MSP[\mathcal{M}]$ in Section 7.4.1).

In both Interpolated Max-Scoring Passage algorithm implementations homogeneity measure $\mathcal{M}$ controls the reliance on document-based versus passage-based scores. (See Equation 10, page 15). We denote the algorithm versions $IMSP^{[\mathcal{M}]}[basic]$, and $IMSP^{[\mathcal{M}]}[\mathcal{M}]$, respectively.

7.5.1 The $IMSP^{[\mathcal{M}]}[basic]$ algorithm

Figure 7 depicts the performance results of the Interpolated Max-Scoring Passage algorithms, denoted $IMSP^{[\mathcal{M}]}[basic]$. To smooth document and passage language models, we use either Jelinek-Mercer smoothing and set $\lambda_{\mathcal{C}} = 0.5$ (as in Section 7.2), or Bayesian smoothing with Diricihlet priors and set $\mu = 1000$ (see Section 2), following some previous work [46].

We see in Figure 7 that our Interpolated Max-Scoring Passage algorithm is in most cases superior to using the Max-Scoring Passage algorithm ($MaxPsg$) with the standard passage language model. We can observe the following for the 48 relevant comparisons for Jelinek-Mercer smoothing (4 corpora $\times$ 4 homogeneity measures $\times$ 3 passage sizes)

- For the MAP evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior to $MaxPsg$

in about 90% of the cases

- For the p@5 evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior (or equal) to $MaxPsg$ in about 59% of the cases
- For the p@10 evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior to $MaxPsg$ in about 65% of the cases

For Dirichlet smoothing, the superiority of $IMSP^{[\mathcal{M}]}[basic]$ over $MaxPsg$ becomes even more evident

- For the MAP evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior to $MaxPsg$ in about 94% of the cases
- For the p@5 evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior (or equal) to $MaxPsg$ in about 94% of the cases
- For the p@10 evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior to $MaxPsg$ in about 96% of the cases

Similarly to the results in Section 7.4, passages of size 50 result in near optimal performance with respect to other passage sizes, and $MSP[length]$ and $MSP[docPsg]$ are the best performing methods in most cases. When passage size is set to 50, and Dirichlet smoothing (which outperforms Jelinek-Mercer smoothing in most cases) is used, $IMSP^{[length]}[basic]$ and $IMSP^{[docPsg]}[basic]$, are both superior to the standard document-based method ($BaseDoc$) in 50% and 67% of the relevant comparisons (4 corpora $\times$ 3 evaluation metrics), respectively; in many cases (e.g., LA+FR45, FR12), the differences are also statistically significant.

We cannot attribute this superiority solely to cases wherein Interpolated Max-Scoring Passage interpolates the score of Max-Scoring Passage with a score derived from a superior algorithm — $BaseDoc$. Case in point, for FR12, $MaxPsg$ is clearly superior to $BaseDoc$, while for LA+FR45 the reverse holds; however, it is almost always the case that for both corpora, shadows that highlight the best performance with respect to an evaluation metric and smoothing technique, appear in rows corresponding to the $IMSP^{[\mathcal{M}]}[basic]$ algorithms. It is interesting to note that in some cases even the performance for AP89, a corpus considered highly homogeneous, benefits from interpolation with passage evidence (e.g., compare the results for $IMSP^{[docPsg]}[basic]$ and $BaseDoc$ with Dirichlet smoothing on AP89).

FR12

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
Jelinek-Mercer Smoothing									
BaseDoc	22.00	19.05	13.33	22.00	**19.05**	13.33	22.00	**19.05**	13.33
MaxPsg	28.44	19.05	14.76	30.14^{d}	**19.05**	**14.76**	18.10	13.33	**13.81**
IMSP$^{[length]}$[basic]	**29.25**	**20.00**	**15.24**	**30.56** ^{d}	17.14	13.81	**25.35**	16.19	12.86
IMSP$^{[ent]}$[basic]	28.07_{p}	19.05	**15.24**	29.96	17.14	14.29	19.61	16.19	13.33
IMSP$^{[docPsg]}$[basic]	29.01	19.05	**15.24**	30.08	17.14	13.81	22.53	15.24	12.86
IMSP$^{[interPsg]}$[basic]	28.82	18.10	14.76	29.99	17.14	14.29	22.54	15.24	12.86
Dirichlet Smoothing									
BaseDoc	25.29	**22.86**	**18.10**	25.29	**22.86**	18.10	25.29	**22.86**	**18.10**
MaxPsg	29.17	17.14	14.76	30.67	**22.86**	17.14	19.23	17.14	14.29
IMSP$^{[length]}$[basic]	**29.97**	20.00	16.67	**30.84**	**22.86**	18.10	$\mathbf{29.81}_{p}$	**22.86**	**18.10**
IMSP$^{[ent]}$[basic]	29.19	20.95	16.67	30.47	21.90	**18.57**	28.99_{p}	20.95	**18.10**
IMSP$^{[docPsg]}$[basic]	29.93	20.95	17.62	30.28	21.90	17.14	29.22_{p}	20.95	17.14
IMSP$^{[interPsg]}$[basic]	29.83	17.14	17.62	30.33	21.90	16.67	28.95	20.95	16.67

LA+FR45

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
Jelinek-Mercer Smoothing									
BaseDoc	**22.69**	**30.64**	26.38	22.69	**30.64**	**26.38**	**22.69**	**30.64**	26.38
MaxPsg	21.93	27.66	25.53	21.68	28.51	25.74	21.70	29.36	**26.81**
IMSP$^{[length]}$[basic]	22.63_{p}	27.66	**26.60**	**22.75** $_{p}$	28.09	24.89	22.36	26.81	24.89
IMSP$^{[ent]}$[basic]	22.08	27.23	25.53	21.80	29.36	25.96	21.79	30.21	26.17
IMSP$^{[docPsg]}$[basic]	22.66	28.51	**26.60**	22.65	25.96	25.32	20.96	26.81	23.62_{p}
IMSP$^{[interPsg]}$[basic]	22.42	28.51	26.17	21.87	27.23	24.89	20.80_{p}	27.66	24.04_{p}
Dirichlet Smoothing									
BaseDoc	24.05	32.34	26.81	24.05	32.34	26.81	24.05	32.34	26.81
MaxPsg	22.50	30.21	26.60	22.25	30.64	26.60	21.96	28.51	26.81
IMSP$^{[length]}$[basic]	24.52_{p}	33.19	26.81	24.89^{d}_{p}	**33.62**	**27.45**	$\mathbf{24.84}^{d}_{p}$	**34.47** $^{d}_{p}$	27.02
IMSP$^{[ent]}$[basic]	23.53_{p}	32.34	26.60	24.74_{p}	32.34	27.23	24.35_{p}	31.49	**27.23**
IMSP$^{[docPsg]}$[basic]	24.44^{d}_{p}	**33.62**	**27.66**	24.76^{d}_{p}	**33.62**	27.02	24.70_{p}	31.91	26.81
IMSP$^{[interPsg]}$[basic]	$\mathbf{24.61}_{p}$	32.34	26.81	**25.02** $_{p}$	31.06	27.02	24.53_{p}	30.21	27.02

Figure continues on the next page.

WSJ

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
Jelinek-Mercer Smoothing									
BaseDoc	28.38	42.40	39.60	**28.38**	42.40	39.60	**28.38**	**42.40**	**39.60**
MaxPsg	28.80	**46.00**	41.80	26.10^{d}	44.00	40.40	24.95^{d}	40.80	37.20
IMSP$^{[length]}$[basic]	**29.25**$_{p}$	**46.00**	42.00	27.43^{d}_{p}	**44.80**	41.40	26.04^{d}_{p}	40.80	37.60
IMSP$^{[ent]}$[basic]	29.16_{p}	45.20	**42.40**	26.68^{d}_{p}	43.60	40.40	25.30^{d}_{p}	41.20	36.40
IMSP$^{[docPsg]}$[basic]	29.17^{d}_{p}	45.20	41.60	27.70_{p}	**44.80**	**42.00**$_{p}$	25.83^{d}_{p}	39.60	37.00
IMSP$^{[interPsg]}$[basic]	29.22_{p}	**46.00**	42.20	27.30_{p}	43.60	41.20	25.44^{d}_{p}	40.00	35.60^{d}
Dirichlet Smoothing									
BaseDoc	32.50	53.60	**48.40**	32.50	53.60	**48.40**	**32.50**	53.60	**48.40**
MaxPsg	31.12^{d}	49.20^{d}	46.00	27.86^{d}	44.80^{d}	41.20^{d}	26.02^{d}	40.00^{d}	37.00^{d}
IMSP$^{[length]}$[basic]	32.58_{p}	53.60_{p}	48.00_{p}	32.08_{p}	**54.80**$_{p}$	48.00_{p}	32.25_{p}	**54.40**$_{p}$	47.60_{p}
IMSP$^{[ent]}$[basic]	32.13_{p}	52.80_{p}	48.20_{p}	31.42_{p}	52.80_{p}	47.00_{p}	31.33^{d}_{p}	**54.40**$_{p}$	47.60_{p}
IMSP$^{[docPsg]}$[basic]	**32.86**$^{d}_{p}$	53.60_{p}	48.20_{p}	**32.55**$_{p}$	**54.80**$_{p}$	47.80_{p}	32.40_{p}	**54.40**$_{p}$	47.40_{p}
IMSP$^{[interPsg]}$[basic]	32.72_{p}	**54.40**$_{p}$	**48.40**$_{p}$	31.86_{p}	52.40_{p}	47.60_{p}	31.22^{d}_{p}	52.40_{p}	46.60_{p}

AP89

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
Jelinek-Mercer Smoothing									
BaseDoc	**19.98**	25.65	**24.13**	**19.98**	25.65	**24.13**	**19.98**	**25.65**	**24.13**
MaxPsg	18.82^{d}	27.83	23.04	17.71^{d}	26.09	22.39	16.34^{d}	20.43	16.52^{d}
IMSP$^{[length]}$[basic]	18.99_{p}	**28.26**	23.04	18.38^{d}_{p}	26.52	23.04	17.15^{d}_{p}	21.30	19.57^{d}_{p}
IMSP$^{[ent]}$[basic]	18.95^{d}_{p}	27.39	23.26	18.00^{d}_{p}	26.52	22.39	16.73^{d}	20.43^{d}	17.83^{d}
IMSP$^{[docPsg]}$[basic]	19.43_{p}	25.65	23.48	18.71^{d}_{p}	**27.39**	22.61	17.25^{d}_{p}	21.74	19.78^{d}_{p}
IMSP$^{[interPsg]}$[basic]	19.04_{p}	26.09	23.04	18.14^{d}_{p}	25.22	22.61	16.75^{d}_{p}	20.87	18.26^{d}
Dirichlet Smoothing									
BaseDoc	20.63	28.26	**26.52**	20.63	28.26	**26.52**	**20.63**	**28.26**	**26.52**
MaxPsg	19.99	25.65	24.35	18.67^{d}	24.35	22.83	17.16^{d}	21.74	18.04^{d}
IMSP$^{[length]}$[basic]	20.37_{p}	26.09	24.78	20.15_{p}	26.96	25.00	20.20^{d}_{p}	26.09	25.00^{d}_{p}
IMSP$^{[ent]}$[basic]	20.22_{p}	26.52	24.57	19.95_{p}	28.26	25.65_{p}	19.87^{d}_{p}	26.52	26.09_{p}
IMSP$^{[docPsg]}$[basic]	**20.71**	**29.57**	25.87	**20.67**$_{p}$	**28.70**	26.09	20.58_{p}	27.83	25.65_{p}
IMSP$^{[interPsg]}$[basic]	20.56	28.70	25.00	20.31_{p}	27.83	25.43	20.10^{d}_{p}	23.04^{d}	24.57_{p}

Figure 7: **Performance numbers of the Interpolated Max-Scoring Passage algorithm ($IMSP^{[\mathcal{M}]}[basic]$).**

$IMSP^{[\mathcal{M}]}[basic]$ interpolates between the document-based language model ($BaseDoc$) score with the score assigned by the Max-Scoring Passage algorithm when implemented with the standard passage language model ($MaxPsg$), using the homogeneity measure $\mathcal{M}$. Either Jelinek-Mercer or Dirichlet smoothing is used for the construction of the language models. Standard document-based ($BaseDoc$) and standard passage-based ($MaxPsg$) language-model retrieval performance is presented for reference. Boldface indicates the best result per column; shadow marks the best performance in a table with respect to an evaluation measure; d and p mark statistically significant differences with $BaseDoc$ and $MaxPsg$ respectively.

Further analysis Analogously to Section 7.4, we now examine the alternative of fixing the balance between the document-based and passage-based query-similarity scores, making the assumption that all documents in the corpus are homogeneous to the same extent, by setting $h^{[\mathcal{M}]}(d)$ to a fixed value for all $d \in \mathcal{C}$. Note that doing so results in fixed interpolation weights, which is reminiscent of Callan's best performing retrieval model [7]. Furthermore, observe that setting $h^{[\mathcal{M}]}(d)$ to 0 or 1 and using the Interpolated Max-Scoring Passage algorithm amounts to using $MaxPsg$ or $BaseDoc$ respectively. (Refer to Section 4.1).

Figure 8 depicts the MAP performance curve of Interpolated Max-Scoring Passage (with passage sizes set to 150, 50 and 25) when setting $h^{[\mathcal{M}]}(d)$ to a fixed value in $\{0, 0.1, \ldots, 1.0\}$. We also plot for comparison the performance of our best performing methods $IMSP^{[docPsg]}[basic]$ and $IMSP^{[length]}[basic]$, with thick and thin horizontal line respectively.

We can see in Figure 8 that for larger passage sizes (passages of size 150 and 50) using the homogeneity measures helps us at the very least to avoid a relatively poor performance obtained by a bad choice of a constant $h^{[\mathcal{M}]}(d)$ (as is the case on AP89). In some cases (see FR12 or WSJ) using homogeneity measures for passages of sizes 150 and 50 yields near (or even better than) optimal performance with respect to that obtained by any fixed weight in $\{0, 0.1, \ldots, 1\}$. It is also important to note that while the performance differences are small in absolute terms, many of them are statistically significant.

- For passage size 150, $IMSP^{[length]}[basic]$'s performance is better to a statistically significant degree than that resulting from setting $h^{[\mathcal{M}]}(d) = 0$ for all tested corpora except FR12, $h^{[\mathcal{M}]}(d) \in \{0.1, \ldots, 0.4\}$ for LA+FR45, $h^{[\mathcal{M}]}(d) \in \{0.1, 0.3\}$ for WSJ and $h^{[\mathcal{M}]}(d) = 0.9$ for FR12. Performance of $IMSP^{[docPsg]}[basic]$ is better to a statistically significant degree than that resulting from setting $h^{[\mathcal{M}]}(d) = 0$ for all corpora and $h^{[\mathcal{M}]}(d) = 0$ for WSJ, LA+FR45 and AP89.

- For passage size 50, $IMSP^{[length]}[basic]$'s performance is better to a statistically significant degree than that resulting from setting $h^{[\mathcal{M}]}(d) = 0$ for all tested corpora except FR12, $h^{[\mathcal{M}]}(d) = 0.1$ for AP89 and $h^{[\mathcal{M}]}(d) = 0.2$ for WSJ. Performance of $IMSP^{[docPsg]}[basic]$ is better to a statistically significant degree than that resulting from setting $h^{[\mathcal{M}]}(d) = 0$ for all tested corpora except FR12, $h^{[\mathcal{M}]}(d) = 0.1$ for AP89, $h^{[\mathcal{M}]}(d) = 0.2$ for WSJ and $h^{[\mathcal{M}]}(d) = 0.6$ for LA+FR45.

Figure 8 also illustrates the performance degradation for passages of size

25, which is in line with our findings in Section 7.4.

We therefore draw the conclusion — which is analogous to the one in Section 7.4 — that the homogeneity measures help to integrate document-based and passage-based similarity scores to result in performance that is often superior to that resulting from using each separately, or resulting from a poor choice of fixed interpolation coefficients in Equation 10 (page 15).

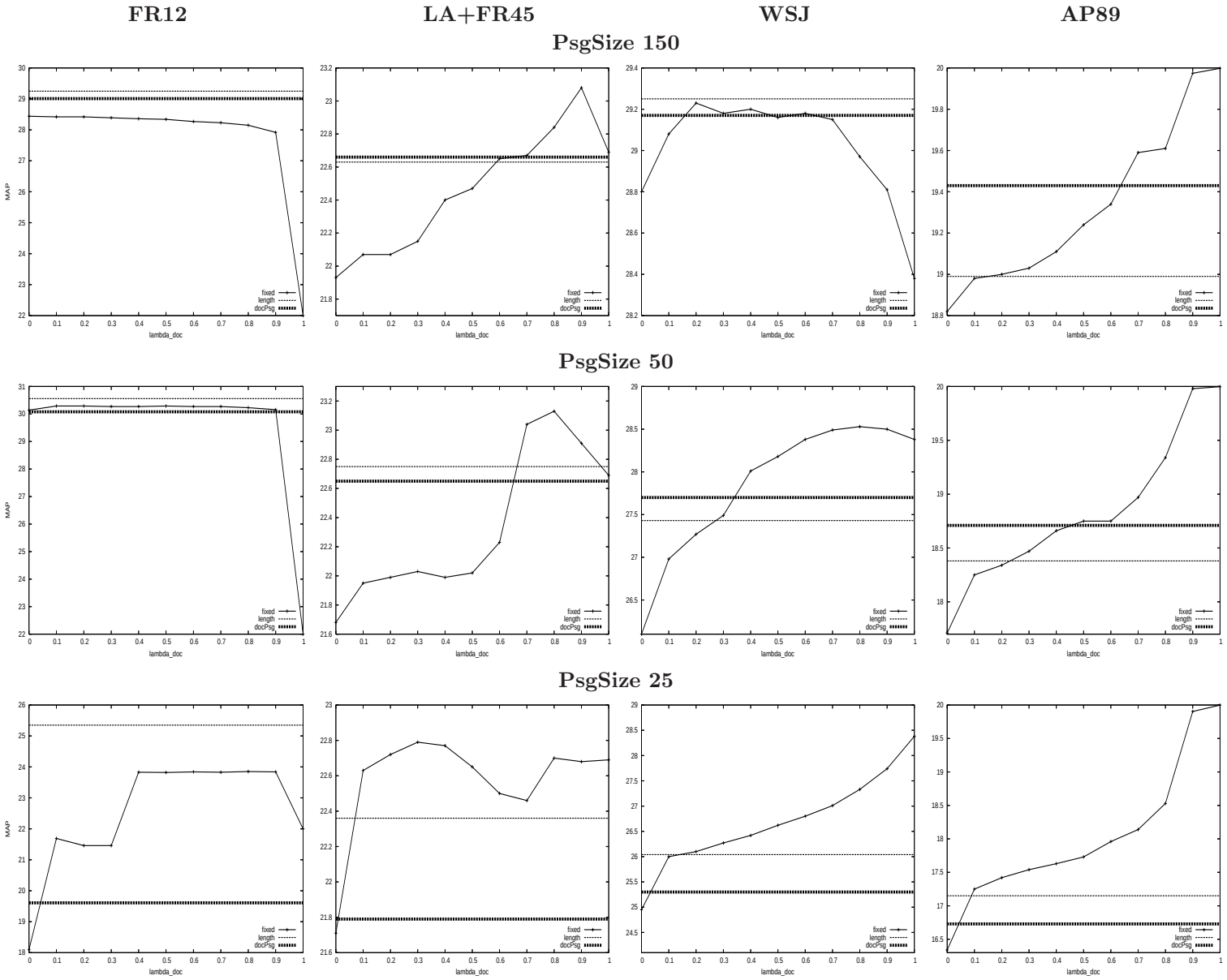

Figure 8: **The Interpolated Max-Scoring Passage algorithm's MAP performance when either setting $h^{[\mathcal{M}]}(d)$ to fixed values or using homogeneity measures.**
The performance is shown when either $h^{[\mathcal{M}]}(d) \in \{0, 0.1, \ldots, 1.0\}$ for all $d \in \mathcal{C}$ (note that 0 and 1 correspond to $MaxPsg$ and $BaseDoc$ respectively), or using the homogeneity measures $length$ and $docPsg$ (thin and thick horizontal lines, respectively) instead, although these measures do *not* incorporate free parameters. (Lines are used for convenience of comparison.) Note: figures are not to the same scale.

7.5.2 The $IMSP^{[\mathcal{M}]}[\mathcal{M}]$ algorithm

Figure 9 depicts the performance results of the Interpolated Max-Scoring Passage algorithms, denoted $IMSP^{[\mathcal{M}]}[\mathcal{M}]$. To smooth document language models we use Jelinek-Mercer smoothing and set $\lambda_{\mathcal{C}} = 0.5$ (as in Section 7.2). As a passage language model we use our language model from Equation 11 (page 11), wherein we smooth the passage maximum-likelihood estimate with those of its ambient document and the collection.

Thus, $IMSP^{[\mathcal{M}]}[\mathcal{M}]$, balances the document and passage evidence on two levels. The first level is the language model used for selecting the passage with the highest query-similarity score; the second level is the score interpolation between document and passage scores that is performed *after* the passage with the highest query-similarity score has been selected. On both levels, we use the homogeneity measure to balance the document and passage information.

Figure 9 compares the performance of $IMSP^{[\mathcal{M}]}[\mathcal{M}]$ to that of our reference comparisons $BaseDoc$ and $MaxPsg$. We see in Figure 9 that $IMSP^{[\mathcal{M}]}[\mathcal{M}]$ is superior to both using the standard document language model ($BaseDoc$) and the Max-Scoring Passage algorithm ($MaxPsg$) with the standard passage language model.

We can observe the following for the 48 relevant comparisons for Jelinek-Mercer smoothing(4 corpora $\times$ 4 homogeneity measures $\times$ 3 passage sizes)

- For the MAP evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior to $MaxPsg$ in about 96% of the cases
- For the p@5 evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior to $MaxPsg$ in about 65% of the cases
- For the p@10 evaluation metric, $IMSP^{[\mathcal{M}]}[basic]$ is superior to $MaxPsg$ in about 88% of the cases

Similarly to the results in sections 7.4.1 and 7.5.1 using passages of size 50 results in in near optimal performance with respect to other passage sizes, and $MSP[length]$ and $MSP[docPsg]$ are the best performing methods in most cases. When passage size is set to 50, $IMSP^{[length]}[basic]$ and $IMSP^{[docPsg]}[basic]$ are both superior in 83% of the relevant comparisons (4 corpora $\times$ 3 evaluation metrics) to the standard document-based method, $BaseDoc$; in some cases (e.g., WSJ and FR12), the differences are also statistically significant.

FR12

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	22.00	19.05	13.33	22.00	19.05	13.33	22.00	19.05	13.33
MaxPsg	28.44	19.05	14.76	30.14^{d}	19.05	14.76	18.10	13.33	13.81
IMSP$^{[length]}$[length]	**29.43**^{d}	**20.95**	**16.19**	**32.74**$^{d}_{p}$	**20.95**	16.67^{d}	**26.77**$_{p}$	**20.95**$_{p}$	15.24
IMSP$^{[ent]}$[ent]	29.12^{d}	18.10	**16.19**	30.01^{d}	18.10	**17.14**^{d}	24.33	19.05	**15.71**
IMSP$^{[docPsg]}$[docPsg]	28.20	18.10	**16.19**	31.06^{d}	19.05	15.71	25.30	19.05	13.33
IMSP$^{[interPsg]}$[interPsg]	28.52	19.05	15.24	30.76^{d}	18.10	15.71	25.37	18.10	12.38

LA+FR45

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	22.69	**30.64**	26.38	22.69	**30.64**	**26.38**	22.69	**30.64**	26.38
MaxPsg	21.93	27.66	25.53	21.68	28.51	25.74	21.71	29.36	**26.81**
IMSP$^{[length]}$[length]	**23.16**$_{p}$	29.79	**27.87**	**24.23**$_{p}$	28.51	25.74	**23.02**	25.96	24.47
IMSP$^{[ent]}$[ent]	22.33	28.94	27.45_{p}	21.80	28.94	25.74	21.73	29.79	25.74
IMSP$^{[docPsg]}$[docPsg]	22.99	29.79	26.81	23.22	27.23	25.11	20.84	25.53	23.40_{p}
IMSP$^{[interPsg]}$[interPsg]	23.00_{p}	29.36	27.02	21.93	26.38	25.74	20.75	28.51	23.19_{p}

WSJ

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	28.38	42.40	39.60	28.38	42.40	39.60	**28.38**	42.40	39.60
MaxPsg	28.80	**46.00**	41.80	26.10^{d}	44.00	40.40	24.95^{d}	40.80	37.20
IMSP$^{[length]}$[length]	29.21^{d}	44.80	41.80	29.05_{p}	45.60	**44.80**$^{d}_{p}$	28.22_{p}	**43.60**	**43.20**$_{p}$
IMSP$^{[ent]}$[ent]	**29.27**^{d}	44.00	42.00	27.87_{p}	44.40	42.40_{p}	26.49^{d}_{p}	41.60	39.60
IMSP$^{[docPsg]}$[docPsg]	29.19^{d}	44.00	41.80^{d}	**29.07**$_{p}$	**46.40**	43.40^{d}_{p}	27.72_{p}	42.80	42.40_{p}
IMSP$^{[interPsg]}$[interPsg]	29.09^{d}	45.20	**42.20**^{d}	28.08_{p}	44.80	43.60^{d}_{p}	26.70^{d}_{p}	42.00	39.00

AP89

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	**19.98**	25.65	**24.13**	**19.98**	25.65	24.13	**19.98**	**25.65**	**24.13**
MaxPsg	18.82^{d}	**27.83**	23.04	17.71^{d}	26.09	22.39	16.34^{d}	20.43	16.52^{d}
IMSP$^{[length]}$[length]	19.36_{p}	26.09	23.04	18.88_{p}	26.96	**24.78**	17.86^{d}_{p}	25.22_{p}	21.74_{p}
IMSP$^{[ent]}$[ent]	19.17_{p}	27.39	22.61	18.23^{d}_{p}	26.09	23.26	17.36^{d}_{p}	20.87	19.13^{d}_{p}
IMSP$^{[docPsg]}$[docPsg]	19.74_{p}	26.09	23.48	19.41_{p}	**27.83**	24.57	17.82^{d}_{p}	**25.65**$_{p}$	21.96_{p}
IMSP$^{[interPsg]}$[interPsg]	19.52_{p}	25.22	23.04	18.38^{d}_{p}	25.22	23.04	17.21^{d}_{p}	23.91_{p}	19.13^{d}_{p}

Figure 9: **Performance numbers of the Interpolated Max-Scoring Passage algorithm ($IMSP^{[\mathcal{M}]}[\mathcal{M}]$).**
$IMSP^{[\mathcal{M}]}[\mathcal{M}]$ interpolates the document-based language model ($BaseDoc$) score with the score assigned by the Max-Scoring Passage algorithm when implemented with our homogeneity-based passage language model ($MSP[\mathcal{M}]$), using the homogeneity measure $\mathcal{M}$. Standard document-based ($BaseDoc$) and standard passage-based ($MaxPsg$) language-model retrieval performance is presented for reference. Boldface indicates the best result per column; shadow marks the best performance in a table with respect to an evaluation measure; d and p mark statistically significant differences with $BaseDoc$ and $MaxPsg$ respectively.

Further analysis We now present a comparison between several of the algorithms proposed in this thesis.

We want to explore whether the Interpolated Max-Scoring Passage algorithm, which assigns a score to a document by the interpolation of the document query-similarity score and the highest query-similarity score of any of its passages, may yield further performance improvements over using Max-Scoring Passage algorithm with our language model $p_g^{[\mathcal{M}]}(\cdot)$, which assigns a score to a document by the highest query-similarity score of any of its passages. Accordingly, we compare in Figure 10 the performance of the $MSP[\mathcal{M}]$ instantiation of the Max-Scoring Passage algorithm from Section 7.4, the $IMSP^{[\mathcal{M}]}[basic]$ instantiation of the Interpolated Max-Scoring Passage algorithm from Section 7.5.1 and the $IMSP^{[\mathcal{M}]}[\mathcal{M}]$ instantiation of the Interpolated Max-Scoring Passage algorithm discussed in the current section.

The main conclusion we can draw from Figure 10 is that the differences between $IMSP^{[\mathcal{M}]}[\mathcal{M}]$, $IMSP^{[\mathcal{M}]}[basic]$ and $MSP[\mathcal{M}]$ performances are rarely statistically significant. There is not a clear indication of superiority of neither of the algorithms; however, generally $IMSP^{[\mathcal{M}]}[\mathcal{M}]$ demonstrates the best performance among the three: out of 33 shadow-marked results (best performance per homogeneity measure) 17 are achieved by $IMSP^{[\mathcal{M}]}[\mathcal{M}]$, 4 are achieved by $IMSP^{[\mathcal{M}]}[basic]$ and 13 are achieved by $MSP[\mathcal{M}]$. However, as mentioned above, in most cases the differences are not statistically significant.

FR12

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
	$h^{[length]}(d)$								
MSP[length]	**29.56**	18.10	15.71	31.83	20.00	15.71	**26.87**	**21.90**	**16.19**
IMSP$^{[length]}$[basic]	29.25^{m}	20.00	15.24	30.56^{m}	17.14	13.81	25.35	16.19	12.86
IMSP$^{[length]}$[length]	29.43	**20.95**	**16.19**	**32.74**$^{m}_{i}$	**20.95**	**16.67**	26.77	20.95	15.24
	$h^{[docPsg]}(d)$								
MSP[docPsg]	**29.32**	**19.05**	**16.19**	31.01	**19.05**	**15.71**	**25.32**	18.10	12.86
IMSP$^{[docPsg]}$[basic]	29.01	**19.05**	15.24	30.08	17.14	13.81	22.53	15.24	12.86
IMSP$^{[docPsg]}$[docPsg]	28.20	18.10	**16.19**	**31.06**	**19.05**	**15.71**	25.30	**19.05**	**13.33**

LA+FR45

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
	$h^{[length]}(d)$								
MSP[length]	23.05	28.94	27.45	23.56	**28.94**	**25.96**	**23.21**	25.53	**25.11**
IMSP$^{[length]}$[basic]	22.63	27.66	26.60	22.75	28.09	24.89	22.36	**26.81**	24.89
IMSP$^{[length]}$[length]	**23.16**	**29.79**	**27.87**	**24.23**	28.51	25.74	23.02	25.96	24.47
	$h^{[docPsg]}(d)$								
MSP[docPsg]	**23.16**	29.36	**27.87**	22.99	26.38	**25.53**	**21.75**	26.38	**24.26**
IMSP$^{[docPsg]}$[basic]	22.66^{m}	28.51	26.60	22.65	25.96	25.32	20.96	**26.81**	23.62
IMSP$^{[docPsg]}$[docPsg]	22.99	**29.79**	26.81	**23.22**	**27.23**	25.11	20.84	25.53	23.40

Figure continues on the next page.

WSJ

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
	$h^{[length]}(d)$								
MSP[length]	**29.25**	44.40	**43.00**	29.00	**46.00**	**44.80**	27.91	**44.00**	**43.40**
IMSP$^{[length]}$[basic]	**29.25**	**46.00**	42.00	27.43^{m}	44.80	41.40^{m}	26.04^{m}	40.80	37.60^{m}
IMSP$^{[length]}$[length]	29.21	44.80	41.80^{m}	**29.05**$_{i}$	45.60	**44.80** $_{i}$	**28.22**$_{i}$	43.60	43.20$_{i}$
	$h^{[docPsg]}(d)$								
MSP[docPsg]	29.13	44.40	**42.60**	**29.15**	45.60	**44.80**	**27.80**	42.00	**42.40**
IMSP$^{[docPsg]}$[basic]	29.17	**45.20**	41.60	27.70^{m}	44.80	42.00^{m}	25.83^{m}	39.60	37.00^{m}
IMSP$^{[docPsg]}$[docPsg]	**29.19**	44.00	41.80	29.07$_{i}$	**46.40**	43.40	27.72$_{i}$	**42.80**	**42.40**$_{i}$

AP89

	PSG 150			PSG 50			PSG 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
	$h^{[length]}(d)$								
MSP[length]	19.32	27.83	**23.70**	18.74	26.09	24.57	17.79	24.78	20.87
IMSP$^{[length]}$[basic]	18.99^{m}	**28.26**	23.04	18.38^{m}	26.52	23.04	17.15^{m}	21.30^{m}	19.57
IMSP$^{[length]}$[length]	**19.36** $^{m}_{i}$	26.09	23.04	**18.88**$^{m}_{i}$	**26.96**	**24.78**	**17.86**$_{i}$	**25.22**$_{i}$	**21.74**$_{i}$
	$h^{[docPsg]}(d)$								
MSP[docPsg]	**19.75**	**26.09**	23.26	19.06	**29.13**	**24.57**	17.73	24.78	**21.96**
IMSP$^{[docPsg]}$[basic]	19.43	25.65	**23.48**	18.71^{m}	27.39	22.61	17.25^{m}	21.74^{m}	19.78^{m}
IMSP$^{[docPsg]}$[docPsg]	19.74	**26.09**	**23.48**	**19.41**$^{m}_{i}$	27.83	**24.57**	**17.82**$_{i}$	**25.65**	**21.96**

Figure 10: **Comparison between the Max-Scoring Passage and Interpolated Max-Scoring Passage algorithms.**
Max-Scoring Passage ($MSP[\mathcal{M}]$) algorithm is instantiated using our language model $p_g^{[\mathcal{M}]}(\cdot)$ and the Interpolated Max-Scoring Passage algorithm is instantiated using either standard passage language model $p_g^{[basic]}(\cdot)$ ($IMSP^{[\mathcal{M}]}[basic]$) or our language model $p_g^{[\mathcal{M}]}(\cdot)$ ($IMSP^{[\mathcal{M}]}[\mathcal{M}]$). Homogeneity measures $h^{[length]}(d)$ and $h^{[docPsg]}(d)$ are used. Boldface indicates the best result per column; shadow marks the best performance per homogeneity measure; m and i mark statistically significant performance differences with respect to $MSP[\mathcal{M}]$ and $IMSP^{[\mathcal{M}]}[basic]$ respectively

7.5.3 Conclusions

In this section we have instantiated two versions of Interpolated Max-Scoring Passage algorithms, the first ($IMSP^{[\mathcal{M}]}[basic]$) using the standard passage language model $p_g^{[basic]}(\cdot)$, and the second ($IMSP^{[\mathcal{M}]}[\mathcal{M}]$) using our passage language model $p_g^{[\mathcal{M}]}(\cdot)$. Both instantiations demonstrated (in many cases significant) performance improvements over both document retrieval ($BaseDoc$) and passage retrieval ($MaxPsg$) baselines. However, incorporating passage language model $p_g^{[\mathcal{M}]}(\cdot)$ into the Interpolated Max-Scoring Passage instantiation $IMSP^{[\mathcal{M}]}[\mathcal{M}]$ did not yield consistent performance improvements over those already achieved by instantiation of the Max-Scoring Passage algorithm using passage language model $p_g^{[\mathcal{M}]}(\cdot)$.

7.6 The Representation-Based Scoring algorithm

The Representation-Based Scoring algorithm determines a score of the document by the query-similarity score of its representation (see Equation 13, page 21). Document representation is selected following the selection principle detailed in Section 6.2; namely, we select a passage that bears the closest similarity to the document as a whole, and use it as a basis for the document representation. In our experiments, we determine the document-passage similarity by means of calculating the cosine measure between the tf.idf vector-space representations [36] of the passage and its ambient document. The passage for which this cosine measure is maximized is selected as the basis for the document representation.

The performance of the Representation-Based Scoring algorithm ($Rep[\mathcal{M}]$) is shown in Figure 11. We can see that in the majority of the cases, performance of Representation-Based Scoring algorithm is inferior to that of our reference comparisons $BaseDoc$ and $MaxPsg$.

One observation to be made is that for FR12 — collection considered as highly heterogeneous [28] — the Representation-Based Scoring algorithm performance is at its worst, showing considerable degradation with respect to reference comparisons; on the other hand, for AP89, which considered to be homogeneous, the Representation-Based Scoring algorithm performance is much better, even outperforming the $MaxPsg$ reference in some cases. This can be attributed to the way the document representation is selected in our experiments. Supposedly, the more heterogeneous the document is, the less is the similarity between itself and its passage-based representation. Thus, we can assume that for heterogeneous documents more elaborated representations should be developed, as a single passage does not quite capture all the information contained in the document; on the other hand, for the case of more homogeneous documents, a single passage can serve as a basis for document representation reasonably well, because of its similarity to a document as a whole.

The main conclusion we draw from experiments with the Representation-Based Scoring algorithm is that the query-similarity consideration is vital in the selection of the representative passage for a document. Performance results from Section 7.4 and Section 7.5 in conjunction with the performance results in the current section show that selection of a document representation according to a query (as in Max-Scoring Passage and Interpolated Max-Scoring Passage algorithms, when selecting a passage with the highest query-similarity score as a basis for document representation) usually leads to a better performance when compared with the selection of a query-

independent document representation (as in the Representation-Based Scoring algorithm).

FR12

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	22.00	**19.05**	13.33	22.00	**19.05**	13.33	**22.00**	**19.05**	13.33
MaxPsg	**28.44**	**19.05**	**14.76**	**30.14** ^{d}	**19.05**	**14.76**	18.10	13.33	**13.81**
Rep[length]	12.52^{d}_{p}	15.24	11.90	11.10^{d}_{p}	13.33	11.43	18.00	14.29	11.90
Rep[ent]	15.07_{p}	14.29	11.43	12.62^{d}_{p}	11.43	10.00_{p}	20.21	14.29	10.95
Rep[docPsg]	14.50_{p}	16.19	12.86	10.78^{d}_{p}	12.38	11.43	17.80	14.29	10.95
Rep[interPsg]	12.11^{d}_{p}	14.29	11.43	9.66^{d}_{p}	11.43	9.52^{d}_{p}	16.33^{d}	12.38	10.95

LA+FR45

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	**22.69**	**30.64**	**26.38**	**22.69**	**30.64**	**26.38**	**22.69**	**30.64**	26.38
MaxPsg	21.93	27.66	25.53	21.68	28.51	25.74	21.71	29.36	**26.81**
Rep[length]	21.46^{d}	29.36	25.11	21.82^{d}	29.36	24.89	22.41	29.36	26.17
Rep[ent]	19.79^{d}_{p}	26.38^{d}	23.83	18.29^{d}_{p}	28.94	24.04	21.62	28.09	23.62
Rep[docPsg]	21.89^{d}	28.94	25.74	21.08^{d}	28.09	24.68	20.82^{d}	26.81	25.11
Rep[interPsg]	21.04^{d}	28.09	25.11	19.57^{d}	27.66	23.62	18.46^{d}_{p}	25.53	23.40

WSJ

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	28.38	42.40	39.60	**28.38**	42.40	39.60	**28.38**	42.40	**39.60**
MaxPsg	**28.80**	**46.00**	**41.80**	26.10^{d}	**44.00**	**40.40**	24.95^{d}	40.80	37.20
Rep[length]	26.38^{d}_{p}	39.60_{p}	40.40	25.51^{d}	40.80	38.00	25.70^{d}	**42.80**	37.40
Rep[ent]	22.96^{d}_{p}	40.40	40.00	19.56^{d}_{p}	34.40^{d}_{p}	32.60^{d}_{p}	18.59^{d}_{p}	35.20^{d}	34.00^{d}
Rep[docPsg]	27.11^{d}_{p}	40.00_{p}	41.60^{d}	24.62^{d}	40.00	37.40	22.64^{d}	40.40	37.40
Rep[interPsg]	24.87^{d}_{p}	38.00^{d}_{p}	40.20	21.27^{d}_{p}	36.00^{d}_{p}	34.40^{d}_{p}	18.57^{d}_{p}	35.60^{d}	34.40^{d}

AP89

	PsgSize 150			PsgSize 50			PsgSize 25		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	**19.98**	25.65	**24.13**	**19.98**	25.65	**24.13**	**19.98**	25.65	**24.13**
MaxPsg	18.82^{d}	**27.83**	23.04	17.71^{d}	**26.09**	22.39	16.34^{d}	20.43	16.52^{d}
Rep[length]	19.36^{d}	23.04_{p}	22.17	17.90^{d}	23.48	22.83	17.00^{d}_{p}	**26.09**	23.26_{p}
Rep[ent]	17.10^{d}_{p}	20.43^{d}_{p}	20.65^{d}	14.61^{d}	23.04	20.00^{d}	13.24^{d}	24.78	22.83_{p}
Rep[docPsg]	19.47	23.48	22.61	17.80^{d}	24.35	22.17	15.98^{d}	**26.09**	23.48_{p}
Rep[interPsg]	19.04^{d}	23.04_{p}	22.39	16.11^{d}	23.04	22.17	12.35^{d}	24.78	22.17_{p}

Figure 11: **Performance numbers of the Representation-Based Scoring algorithm ($Rep[\mathcal{M}]$).**
Algorithm is implemented using Equation 13 (page 21). Standard document-based ($BaseDoc$) and standard passage-based ($MaxPsg$) language-model retrieval performance is presented for reference. Boldface indicates the best result per column; shadow marks the best performance in a table with respect to an evaluation measure; d and p mark statistically significant differences with $BaseDoc$ and $MaxPsg$ respectively.

7.7 Evaluation summary

In this section we summarize the experimental results presented in this chapter, and draw conclusions about the performance of our algorithms' instantiations. To this end, we refer back to Figure 1, which presents the summary of all algorithms instantiations used in our experiments. In order to compare the different algorithms used, we choose to present performance results for passages of size 50, which show near optimal performance with respect to other passage sizes for both our methods and the reference comparisons. We present results for the *length* homogeneity model. The performance results for all collections for each of the methods in Figure 1 are presented in Figure 12 below.

It is clear from Figure 12 that some of the methods we propose (namely $MSP[\mathcal{M}]$, $IMSP^{[\mathcal{M}]}[basic]$, $IMSP^{[\mathcal{M}]}[\mathcal{M}]$ and $RelPsg[\mathcal{M}]$) are superior to their respective reference comparisons $BaseDoc$, $MaxPsg$, $RelDoc$ and $RelPsg$. In some of the cases, the differences are statistically significant.

An algorithm that consistently shows the best performance among all our algorithms is $RelPsg[\mathcal{M}]$, which uses passage-based *relevance models* (see Section 7.4.3 for more details). Among our language-model based algorithms, $IMSP^{[\mathcal{M}]}[\mathcal{M}]$, which integrates the $MSP[\mathcal{M}]$ and $IMSP^{[\mathcal{M}]}[basic]$ algorithms (see Section 7.5.2), shows the best performance; it is better than both reference comparisons $BaseDoc$, $MaxPsg$ and the $MSP[\mathcal{M}]$ and $IMSP^{[\mathcal{M}]}[basic]$ algorithms it integrates in 8 out of 12 cases.

As a summary to this chapter, we can draw a conclusion that an array of experiments performed on several TREC corpora unequivocally demonstrates the merits of using our homogeneity-based passage language models for the ad hoc document retrieval task. We showed that document retrieval performance can be (in some cases significantly) improved by using Max-Scoring Passage and Interpolated Max-Scoring Passage algorithms, instantiated using either our homogeneity-based language model (as in $MSP[\mathcal{M}]$, $IMSP^{[\mathcal{M}]}[basic]$ and $IMSP^{[\mathcal{M}]}[\mathcal{M}]$) or our homogeneity-based relevance model (as in $RelPsg[\mathcal{M}]$).

	FR12			LA+FR45			WSJ			AP89		
	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10	MAP	p@5	p@10
BaseDoc	22.00	19.05	13.33	22.69	**30.64**	**26.38**	28.38	42.40	39.60	**19.98**	25.65	24.13
MaxPsg	30.14^{d}	19.05	14.76	21.68	28.51	25.74	26.10^{d}	44.00	40.40	17.71^{d}	26.09	22.39
MeanPsg	13.38^{d}_{p}	7.62^{d}_{p}	6.67^{d}_{p}	16.26^{d}_{p}	19.15^{d}_{p}	18.09^{d}_{p}	14.03^{d}_{p}	21.20^{d}_{p}	19.60^{d}_{p}	14.27^{d}_{p}	14.78^{d}_{p}	13.48^{d}_{p}
MSP[length]	31.83^{d}_{p}	20.00	15.71	23.56_{p}	28.94	25.96	29.00_{p}	**46.00**	$\mathbf{44.80}^{d}_{p}$	18.74_{p}	26.09	24.57
IMSP$^{[length]}$[basic]	30.56^{d}	17.14	13.81	22.75_{p}	28.09	24.89	27.43^{d}_{p}	44.80	41.40	18.38^{d}_{p}	26.52	23.04
IMSP$^{[length]}$[length]	$\mathbf{32.74}^{d}_{p}$	**20.95**	$\mathbf{16.67}^{d}$	$\mathbf{24.23}_{p}$	28.51	25.74	$\mathbf{29.05}_{p}$	45.60	$\mathbf{44.80}^{d}_{p}$	18.88_{p}	**26.96**	**24.78**
Rep[length]	11.10^{d}_{p}	13.33	11.43	21.82^{d}	29.36	24.89	25.51^{d}	40.80	38.00	17.90^{d}	23.48	22.83
RelDoc	10.70	10.48	9.05	20.69	28.09	23.83	33.85	48.80	48.40	**25.56**	31.30	28.48
RelPsg	$\mathbf{31.06}^{d}$	19.05	16.19^{d}	21.87	29.79	24.68	33.97	47.20	45.00	22.18	30.43	25.87
RelPsg[length]	30.74^{d}	$\mathbf{23.81}^{d}$	$\mathbf{18.10}^{d}$	$\mathbf{23.30}^{d}_{p}$	**33.19**	**25.32**	$\mathbf{37.53}^{d}_{p}$	**49.60**	$\mathbf{49.00}_{p}$	24.30_{p}	**33.48**	$\mathbf{30.00}_{p}$

Figure 12: **Summary of the performance results of all the evaluated algorithms.**
Results for passage size 50 and homogeneity model *length* are presented for each collection. Best result in a column is boldfaced; significant differences with *BaseDoc* (or *RelDoc* in case of *RelPsg*[*length*]) and *MaxPsg* (or *RelPsg* in case of *RelPsg*[*length*]) are marked with d and p respectively.

8 Conclusions and Future Work

We have explored the potential benefits in utilizing passage-based information for ad hoc document retrieval. To this end, we presented a general probabilistic model for passage-based ad hoc document retrieval and showed that several of the previously suggested passage-based document ranking approaches [28, 7, 45] can be derived from this probabilistic model.

Some previous work on utilization of passages in various information retrieval tasks proposed an integration of information from a passage with information from its ambient document as means to improve retrieval performance [7, 6, 15, 32, 44]; fixed weights were used to control this integration. Instead, we presented measures for estimating *document homogeneity* and used them for integrating passage information with that of its ambient document. We showed that our homogeneity measures help to both fuse document-based and passage-based ranking of documents, and to derive a new passage language model; this new passage language model is effective for passage-based document ranking and for constructing and utilizing *passage-based relevance models*. In many cases, using our homogeneity measures resulted in near (or even better than) optimal retrieval performance with respect to that of the paradigm of fixing the document-passage information integration weights.

We instantiated several retrieval methods based on our general probabilistic framework, among which are: *(i)* an algorithm that ranks a document by the highest query-similarity score of any of its passages, and *(ii)* an algorithm that ranks a document by an interpolation of the document query-similarity score and the highest query-similarity score of any of its passages. We showed that implementation of these algorithms using our homogeneity-based passage language model results in retrieval performance that is in many cases superior to both document retrieval using standard document language model and implementation of these algorithms using standard passage language model. In addition, we showed that using our homogeneity-based passage language model can be used to construct *passage-based relevance models* that are in many cases more effective than both document *relevance models* [25] and standard passage *relevance models* [28]. In the experiments we performed, our homogeneity-based passage language model was shown to be effective both for heterogeneous and homogeneous corpora.

The performance attained by using our homogeneity-based language model seems very promising and calls for further investigation and generalization. Using relatively simple homogeneity measures, such as document length, for integration of passage information with the information from its

ambient document results in performance comparable to (or even acceding) the best performance attained by using fixed weights for the integration. This indicates that there might be even more potential for improvement if homogeneity schemes more sophisticated than the ones proposed in this thesis were applied. It is important to mention that the best retrieval performance attained by using fixed weights for the integration of information from a passage with information from its ambient document does not impose an upper bound on the performance that could be attained using our homogeneity measures. In the latter case *each* document is assigned an individual homogeneity measure value, while in the former case the same parameter values are used across all documents in the collection, which is equivalent to the assumption that all documents are homogeneous to the same extent.

Our experimental results demonstrate the effectiveness of using passage-based information for ad hoc document retrieval. The retrieval performance attained by a simple fixed-sized *window passages* used in our experiments calls for further examination of our algorithms' performance with more complex passage types, such as *arbitrary passages* [19, 18] of varying lengths.

References

[1] Nasreen Abdul-Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah Larkey, Xiaoyan Li, Marck D. Smucker, and Courtney Wade. UMASS at TREC 2004 — novelty and hard. In *Proceedings of the Thirteenth Text Retrieval Conference (TREC-13)*, 2004.

[2] James Allan. HARD track overview in TREC 2003: High accuracy retrieval from documents. In *Proceedings of the Twelfth Text Retrieval Conference (TREC-12)*, pages 24–37, 2003.

[3] James Allan, Margaret E. Connell, W. Bruce Croft, Fang-Fang Feng, David Fisher, and Xiaoyan Li. INQUERY and TREC-9. In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*, pages 551–562, 2000. NIST Special Publication 500-249.

[4] P. B. Baxendale. Machine-made index for technical literaturean experiment. *Journal of Research and Development*, 2(4):354–361, 1958.

[5] Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic query expansion using SMART: TREC3. In *Proceedings of of the Third Text Retrieval Conference (TREC-3)*, pages 69–80, 1994.

[6] Deng Cai, Shipeng Yu, Ji-Rong Wen, and Wei-Ying Ma. Block-based web search. In *Proceedings of SIGIR*, pages 456–463, 2004.

[7] James P. Callan. Passage-level evidence in document retrieval. In *Proceedings of SIGIR*, pages 302–310, 1994.

[8] Andres Corrada-Emmanuel, W. Bruce Croft, and Vanessa Murdock. Answer passage retrieval for question answering. Technical Report IR-283, Center for Intelligent Information Retrieval, University of Massachusetts, 2003.

[9] W. Bruce Croft and John Lafferty, editors. *Language Modeling for Information Retrieval*. Number 13 in Information Retrieval Book Series. Kluwer, 2003.

[10] Ludovic Denoyer, Hugo Zaragoza, and Patrick Gallinari. HMM-based passage models for document classification and ranking. In *Proceedings of ECIR*, pages 126–135, 2001.

[11] David A. Grossman and Ophir Frieder. *Information Retrieval: Algorithms and Heuristics*. Kluwer, 1998.

[12] Marti A. Hearst and Christian Plaunt. Subtopic structuring for full-length document access. In *Proceedings of SIGIR*, pages 56–89, 1993.

[13] Djoerd Hiemstra and Wessel Kraaij. Twenty-One at TREC7: Ad hoc and cross-language track. In *Proceedings of the Seventh Text Retrieval Conference (TREC-7)*, pages 227–238, 1999.

[14] Xiao Hu, Sindhura Bandhakavi, and ChengXiang Zhai. Error analysis of difficult TREC topics. In *Proceedings of SIGIR*, pages 407–408, 2003. Poster.

[15] Munawar Hussain. Language modeling based passage retrieval for question answering systems. Master's thesis, Saarland University, 2004.

[16] Jing Jiang and Chengxiang Zhai. UIUC in HARD 2004 — passage retrieval using HMMs. In *Proceedings of the Thirteenth Text Retrieval Conference (TREC-13)*, 2004.

[17] Karen Sparck Jones and eds. Peter Willett. *Readings in Information Retrieval*. Morgan Kaufmann, 1997.

[18] Marcin Kaszkiel and Justin Zobel. Passage retrieval revisited. In *Proceedings of SIGIR*, pages 178–185, 1997.

[19] Marcin Kaszkiel and Justin Zobel. Effective ranking with arbitrary passages. *Journal of the American Society for Information Science*, 52(4):344–364, November 2001.

[20] Julian Kupiec, Jan O. Pedersen, and Francine Chen. A trainable document summarizer. In *SIGIR*, pages 68–73, 1995.

[21] Oren Kurland and Lillian Lee. Corpus structure, language models, and ad hoc information retrieval. In *Proceedings of SIGIR*, pages 194–201, 2004.

[22] Oren Kurland and Lillian Lee. PageRank without hyperlinks: Structural re-ranking using links induced by language models. In *Proceedings of SIGIR*, pages 306–313, 2005.

[23] Oren Kurland, Lillian Lee, and Carmel Domshlak. Better than the real thing? Iterative pseudo-query processing using cluster-based language models. In *Proceedings of SIGIR*, pages 19–26, 2005.

[24] John Lafferty and ChengXiang Zhai. Probabilistic relevance models based on document and query generation. In Croft and Lafferty [9], pages 1–10.

[25] Victor Lavrenko and W. Bruce Croft. Relevance-based language models. In *Proceedings of SIGIR*, pages 120–127, 2001.

[26] Victor Lavrenko and W. Bruce Croft. Relevance models in information retrieval. In Croft and Lafferty [9], pages 11–56.

[27] J. Lin, D. Quan, V. Sinha, K. Bakshi, D. Huynh, B. Katz, and D. R. Karger. What makes a good answer? the role of context in question answering. In *Proceedings of the Ninth IFIP TC13 International Conference on Human-Computer Interaction(INTERACT-2003)*, pages 25–32, 2003.

[28] Xiaoyong Liu and W. Bruce Croft. Passage retrieval based on language models. In *Proceedings of the 11th International Conference on Information and Knowledge Managment (CIKM)*, pages 375–382, 2002.

[29] H. P. Luhn. The automatic creation of literature abstracts. *Journal of Research and Development*, 2(2):159–165, 1958.

[30] David R. H. Miller, Tim Leek, and Richard M. Schwartz. A hidden Markov model information retrieval system. In *Proceedings of SIGIR*, pages 214–221, 1999.

[31] Elke Mittendorf and Peter Schäuble. Document and passage retrieval based on hidden Markov models. In *Proceedings of SIGIR*, pages 318–327, 1994.

[32] Vanessa Murdock and W. Bruce Croft. A translation model for sentence retrieval. In *Proceedings of HLT/EMNLP*, pages 684–695, 2005.

[33] Jay M. Ponte and W. Bruce Croft. Text segmentation by topic. In *Proceedings of the European Conference on Research and Advanced Technology for Digital Libraries*, pages 113–125, 1997.

[34] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of SIGIR*, pages 275–281, 1998.

[35] Ronald Rosenfeld. Two decades of statistical language modeling: where do we go from here? *Proceedings of the IEEE*, 88(9), 2000.

[36] Gerard Salton. *Automatic Information Organization and Retrieval*. McGraw-Hill computer science series. McGraw-Hill, New York, 1968.

[37] Gerard Salton, James Allan, and Chris Buckley. Approaches to passage retrieval in full text information systems. In *Proceedings of SIGIR*, pages 49–58, 1993.

[38] Gerard Salton and Chris Buckley. Automatic text structuring and retrieval-experiments in automatic encyclopedia searching. In *Proceedings of SIGIR*, pages 21–30, 1991.

[39] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *Proceedings of SIGIR*, pages 21–29, 1996.

[40] Fei Song and W. Bruce Croft. A general language model for information retrieval (poster abstract). In *Proceedings of SIGIR*, pages 279–280, 1999.

[41] Stefanie Tellex, Boris Katz, Jimmy Lin, Aaron Fernandes, and Gregory Marton. Quantitative evaluation of passage retrieval algorithms for question answering. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 41 – 47, 2003.

[42] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, second edition, 1979.

[43] Ellen M. Voorhees and Donna K. Harman, editors. *The Eighth Text REtrieval Conference (TREC-8)*. NIST, 2000.

[44] Courtney Wade and James Allan. Passage retrieval and evaluation. Technical Report IR-396, Center for Intelligent Information Retrieval (CIIR), University of Massachusetts, 2005.

[45] Ross Wilkinson. Effective retrieval of structured documents. In *Proceedings of SIGIR*, pages 311–317, 1994.

[46] Chengxiang Zhai and John D. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR*, pages 334–342, 2001.

[47] Dell Zhang and Wee Sun Lee. A language modeling approach to passage question answering. In *Proceedings of the Twelfth Text Retrieval Conference (TREC-12)*, pages 489–495, 2004.