Examining the Content Load of Part of Speech Blocks for Information Retrieval

Christina Lioma
Department of Computing Science
University of Glasgow
17 Lilybank Gardens
Scotland, U.K.
xristina@dcs.gla.ac.uk

Iadh Ounis
Department of Computing Science
University of Glasgow
17 Lilybank Gardens
Scotland, U.K.
ounis@dcs.gla.ac.uk

Abstract

We investigate the connection between part of speech (POS) distribution and content in language. We define POS blocks to be groups of parts of speech. We hypothesise that there exists a directly proportional relation between the frequency of POS blocks and their content salience. We also hypothesise that the class membership of the parts of speech within such blocks reflects the content load of the blocks, on the basis that open class parts of speech are more content-bearing than closed class parts of speech. We test these hypotheses in the context of Information Retrieval, by syntactically representing queries, and removing from them content-poor blocks, in line with the aforementioned hypotheses. For our first hypothesis, we induce POS distribution information from a corpus, and approximate the probability of occurrence of POS blocks as per two statistical estimators separately. For our second hypothesis, we use simple heuristics to estimate the content load within POS blocks. We use the Text REtrieval Conference (TREC) queries of 1999 and 2000 to retrieve documents from the WT2G and WT10G test collections, with five different retrieval strategies. Experimental outcomes confirm that our hypotheses hold in the context of Information Retrieval.

1 Introduction

The task of an Information Retrieval (IR) system is to retrieve documents from a collection, in response to a user need, which is expressed in the form of a query. Very often, this task is realised by indexing the documents in the collection with keyword descriptors. Retrieval consists in matching the query against the descriptors of the documents, and returning the ones that appear closest, in ranked lists of relevance (van Rijsbergen, 1979). Usually, the keywords that constitute the document descriptors are associated with individual weights, which capture the importance of the keywords to the content of the document. Such weights, commonly referred to as term weights, can be computed using various term weighting schemes. Not all words can be used as keyword descriptors. In fact, a relatively small number of words accounts for most of a document's content (van Rijsbergen, 1979). Function words make 'noisy' index terms, and are usually ignored during the retrieval process. This is practically realised with the use of stopword lists, which are lists of words to be exempted when indexing the collection and the queries.

The use of stopword lists in IR is a manifestation of a well-known bifurcation in linguistics between open and closed classes of words (Lyons, 1977). In brief, open class words are more content-bearing than closed class words. Generally, the open class contains parts of speech that are morphologically and semantically flexible, while the closed class contains words that primarily perform linguistic well-formedness functions. The membership of the closed class is mostly fixed and largely restricted to function words, which are not prone to semantic or morphological alterations.

We define a block of parts of speech (*POS block*) as a block of fixed length n, where n is set empirically. We define *POS block* tokens as individual instances of *POS blocks*, and *POS block*

Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pages 531–538,
Sydney, July 2006.

types as distinct *POS blocks* in a corpus. The purpose of this paper is to test two hypotheses.

The intuition behind both of these hypotheses is that, just as individual words can be content-rich or content-poor, the same can hold for blocks of parts of speech. According to our first hypothesis, *POS blocks* can be categorized as content-rich or content-poor, on the basis of their distribution within a corpus. Specifically, we hypothesise that the more frequently a *POS block* occurs in language, the more content it is likely to bear. According to our second hypothesis, *POS blocks* can be categorized as content-rich or content-poor, on the basis of the part of speech class membership of their individual components. Specifically, we hypothesise that the more closed class components found in a *POS block*, the less content the block is likely to bear.

Both aforementioned hypotheses are evaluated in the context of IR as follows. We observe the distribution of *POS blocks* in a corpus. We create a list of *POS block* types with their respective probabilities of occurrence. As a first step, to test our first hypothesis, we remove the *POS blocks* with a low probability of occurrence from each query, on the assumption that these blocks are content-poor. The decision regarding the threshold θ of low probability of occurrence is realised empirically. As a second step, we further remove from each query *POS blocks* that contain less open class than closed class components, in order to test the validity of our second hypothesis, as an extension of the first hypothesis. We retrieve documents from two standard IR English test collections, namely WT2G and WT10G. Both of these collections are commonly used for retrieval effectiveness evaluations in the Text REtrieval Conference (TREC), and come with sets of queries and query relevance assessments[1]. Query relevance assessments are lists of relevant documents, given a query. We retrieve relevant documents using firstly the original queries, secondly the queries produced after step 1, and thirdly the queries produced after step 2. We use five statistically different term weighting schemes to match the query terms to the document keywords, in order to assess our hypotheses across a range of retrieval techniques. We associate improvement of retrieval performance with successful noise reduction in the queries. We assume noise reduction to reflect the correct identification of content-poor blocks, in line with our hypotheses.

Section 2 presents related studies in this field. Section 3 introduces our methodology. Section 4 presents the experimental settings used to test our hypotheses, and their evaluation outcomes. Section 5 provides our conclusions and remarks.

2 Related Studies

We examine the distribution of *POS blocks* in language. This is but one type of language distribution analysis that can be realised. One can also examine the distribution of character or word n-grams, e.g. Language Modeling (Croft and Lafferty, 2003), phrases (Church and Hanks, 1990; Lewis, 1992), and so on. In class-based n-gram modeling (Brown et al., 1992) for example, class-based n-grams are used to determine the probability of occurrence of a POS class, given its preceding classes, and the probability of a particular word, given its own POS class. Unlike the class-based n-gram model, we do not use *POS blocks* to make predictions. We estimate their probability of occurrence as blocks, not the individual probabilities of their components, motivated by the intuition that the more frequently a *POS block* occurs, the more content it bears. In the context of IR, efforts have been made to use syntactic information to enhance retrieval (Smeaton, 1999; Strzalkowski, 1996; Zukerman and Raskutti, 2002), but not by using POS block-based distribution representations.

3 Methodology

We present the steps realised in order to assess our hypotheses in the context of IR. Firstly, *POS blocks* with their respective frequencies are extracted from a corpus. The probability of occurrence of each *POS block* is statistically estimated. In order to test our first hypothesis, we remove from the query all but *POS blocks* of high probability of occurrence, on the assumption that the latter are content-rich. In order to test our second hypothesis, *POS blocks* that contain more closed class than open class tags are removed from the queries, on the assumption that these blocks are content-poor.

3.1 Inducing *POS blocks* from a corpus

We extract *POS blocks* from a corpus and estimate their probability of occurrence, as follows.

[1] http://trec.nist.gov/

The corpus is POS tagged. All lexical word forms are eliminated. Thus, sentences are constituted solely by sequences of POS tags. The following example illustrates this point.

> [Original sentence] Many of the proposals for directives and action programmes planned by the Commission have for some obscure reason never seen the light of day.
>
> [Tagged sentence] Many/JJ of/IN the/DT proposals/NNS for/IN directives/NNS and/CC action/NN programmes/NNS planned/VVN by/IN the/DT Commission/NP have/VHP for/IN some/DT obscure/JJ reason/NN never/RB seen/VVN the/DT light/NN of/IN day/NN
>
> [Tags-only sentence] JJ IN DT NNS IN NNS CC NN NNS VVN IN DT NP VHP IN DT JJ NN RB VVN DT NN IN NN

For each sentence in the corpus, all possible *POS blocks* are extracted. Thus, for a given sentence ABCDEFGH, where POS tags are denoted by single letters, and where POS block length n = 4, the *POS blocks* extracted are ABCD, BCDE, CDEF, and so on. The extracted *POS blocks* overlap. The order in which the *POS blocks* occur in the sentence is disregarded.

We statistically infer the probability of occurrence of each *POS block*, on the basis of the individual *POS block* frequencies counted in the corpus. Maximum Likelihood inference is eschewed, as it assigns the maximum possible likelihood to the *POS blocks* observed in the corpus, and no probability to unseen *POS blocks*. Instead, we employ statistical estimation that accounts for unseen *POS blocks*, namely Laplace and Good-Turing (Manning and Schutze, 1999).

3.2 Removing *POS blocks* from the queries

In order to test our first hypothesis, *POS blocks* of low probability of occurrence are removed from the queries. Specifically, we POS tag the queries, and remove the *POS blocks* that have a probability of occurrence below an empirical threshold θ. The following example illustrates this point.

> [Original query] A relevant document will focus on the causes of the lack of integration in a significant way; that is, the mere mention of immigration difficulties is not relevant. Documents that discuss immigration problems unrelated to Germany are also not relevant.
>
> [Tags-only query] DT JJ NN MD VV IN DT NNS IN DT NN IN NN IN DT JJ NN; WDT VBZ DT JJ NN IN NN NNS VBZ RB JJ. NNS WDT VVP NN NNS JJ TO NP VBP RB RB JJ
>
> [Query with high-probability *POS blocks*] DT NNS IN DT NN IN NN IN NN IN NN NNS
>
> [Resulting query] the causes of the lack of integration in mention of immigration difficulties

Some of the low-probability *POS blocks*, which are removed from the query in the above example, are DT JJ NN MD, JJ NN MD VV, NN MD VV IN, and so on. The resulting query contains fragments of the original query, assumed to be content-rich. In the context of the bag-of-words approach to IR investigated here, the grammatical well-formedness of the query is thus not an issue to be considered.

In order to test the second hypothesis, we remove from the queries *POS blocks* that contain less open class than closed class components. We propose a simple heuristic *Content Load* algorithm, to 'count' the presence of content within a *POS block*, on the premise that open class tags bear more content than closed class tags. The order of tags within a *POS block* is ignored. Figure 1 displays our *Content Load* algorithm.

After the n^{th} *POS block* component has been 'counted', if the *Content Load* is zero or more, we consider the *POS block* content-rich. If the

Figure 1: The *Content Load* algorithm

```
function CONTENT-LOAD(POSblock)
returns ContentLoad
INITIALISE-FOR-EACH-POSBLOCK(query)
for pos ← from 1 to POSblock-size do
if(current-tag = = OpenClass)
(ContentLoad)+ +
elseif(current-tag = = ClosedClass)
(ContentLoad)- -
end
return(ContentLoad)
```

Content Load is strictly less than zero, we consider the *POS block* content-poor. We assume an underlying equivalence of content in all open class parts of speech, which albeit being linguistically counter-intuitive, is shown to be effective when applied to IR (Section 4). The following example illustrates this point. In this example, *POS block* length $n = 4$.

> [Original query] A relevant document will focus on the causes of the lack of integration in a significant way; that is, the mere mention of immigration difficulties is not relevant. Documents that discuss immigration problems unrelated to Germany are also not relevant.
>
> [Tags-only query] DT JJ NN MD VV IN DT NNS IN DT NN IN NN IN DT JJ NN; WDT VBZ DT JJ NN IN NN NNS VBZ RB JJ. NNS WDT VVP NN NNS JJ TO NP VBP RB RB JJ
>
> [Query with high-probability POS blocks] DT NNS IN DT NN IN NN IN NN IN NN NNS
>
> [Content Load of *POS blocks*]
> DT NNS IN DT (-2), NN IN NN IN (0), NN IN NN NNS (+2)
>
> [Query with high-probability *POS blocks* of zero or positive Content Load] NN IN NN IN NN IN NN NNS
>
> [Resulting query] lack of integration in mention of immigration difficulties

4 Evaluation

We present the experiments realised to test the two hypotheses formulated in Section 1. Section 4.1 presents our experimental settings, and Section 4.2 our evaluation results.

4.1 Experimental Settings

We induce *POS blocks* from the English language component of the second release of the parallel Europarl corpus(75MB)[2]. We POS tag the corpus using the TreeTagger[3], which is a probabilistic POS tagger that uses the Penn TreeBank tagset (Marcus et al., 1993). Since we are solely interested in a POS analysis, we introduce a stage of tagset simplification, during which, any information on top of surface POS classification is lost (Table 1). Practically, this leads to 48 original TreeBank (TB) tag classes being narrowed down to 15 Reduced TreeBank (RTB) tag classes. Additionally, tag names are shortened into two-letter names, for reasons of computational efficiency. We consider the TBR tags JJ, FW, NN, and VB as open-class, and the remaining tags as closed class (Lyons, 1977). We extract 214,398,227 *POS block* tokens and 19,343 *POS block* types from the corpus.

Table 1: Correspondence between the TreeBank (TB) and Reduced TreeBank (RTB) tags.

TB	TBR
JJ, JJR, JJS	JJ
RB,RBR,RBS	RB
CD, LS	CD
CC	CC
DT, WDT, PDT	DT
FW	FW
MD, VB, VBD, VBG, VBN, VBP, VBZ, VH, VHD, VHG, VHN, VHP, VHZ	MD
NN, NNS, NP, NPS	NN
PP, WP, PP$, WP$, EX, WRB	PP
IN, TO	IN
POS	PO
RP	RP
SYM	SY
UH	UH
VV, VVD, VVG, VVN, VVP, VVZ	VB

We retrieve relevant documents from two standard TREC test collections, namely WT2G (2GB) and WT10G (10GB), from the 1999 and 2000 TREC Web tracks, respectively. We use the queries 401-450 from the ad-hoc task of the 1999 Web track, for the WT2G test collection, and the queries 451-500 from the ad-hoc task of the 2000 Web track, for the WT10G test collection, with their respective relevance assessments. Each query contains three fields, namely *title*, *description*, and *narrative*. The *title* contains keywords describing the information need. The *description* expands briefly on the information need. The *narrative* part consists of sentences denoting key concepts to be considered or ignored. We use all three

[2] http://people.csail.mit.edu/koehn/publications/europarl/

[3] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

query fields to match query terms to document keyword descriptors, but extract *POS blocks* only from the narrative field of the queries. This choice is motivated by the two following reasons. Firstly, the *narrative* includes the longest sentences in the whole query. For our experiments, longer sentences provide better grounds upon which we can test our hypotheses, since the longer a sentence, the more *POS blocks* we can match within it. Secondly, the *narrative* field contains the most noise in the whole query. Especially when using bag-of-words term weighting, such as in our evaluation, information on what is not relevant to the query only introduces noise. Thus, we select the most noisy field of the query to test whether the application of our hypotheses indeed results in the reduction of noise.

During indexing, we remove stopwords, and stem the collections and the queries, using Porter's[4] stemming algorithm. We use the Terrier[5] IR platform, and apply five different weighting schemes to match query terms to document descriptors. In IR, term weighting schemes estimate the relevance $R(d, Q)$ of a document d for a query Q, as: $R(d, Q) = \sum_{t \in Q} qtw \cdot w(t, d)$, where t is a term in Q, qtw is the query term weight, and $w(t, d)$ is the weight of document d for term t. For example, we use the classical TF_IDF weighting scheme (Sparck-Jones, 1972; Robertson et al., 1995): $w(t, d) = tfn \cdot \log_2 \frac{N}{df+1}$, where tfn is the normalised term frequency in a document: $tfn = \frac{k_1 \cdot tf}{tf + k_1(1-b+b\frac{l}{avg\_l})}$; tf is the frequency of a term in a document; $k1$, and b are parameters; l and avg_l are the document length and the average document length in the collection, respectively; N is the number of documents in the collection; and df is the number of documents containing the term t. For all weighting schemes we use, $qtw = \frac{qtf}{qtf_{max}}$, where qtf is the query term frequency, and qtf_{max} is the maximum qtf among all query terms. We also use the well-established probabilistic BM25 weighting scheme (Robertson et al., 1995), and three distinct weighting schemes from the more recent Divergence From Randomness (DFR) framework (Amati, 2003), namely BB2, PL2, and DLH. Note that, even though we use three weighting schemes from the DFR framework, the said schemes are statistically different to one another. Also, DLH is the only parameter-free weighting scheme we use, as it computes all of the $w(t, d)$ variables automatically from the collection statistics.

We use the default values of all parameters, namely, for the TF_IDF and BM25 weighting schemes (Robertson et al., 1995), $k1 = 1.2$, $k3 = 1000$, and $b = 0.75$ for both test collections; while for the PL2 and BB2 term weighting schemes (Amati, 2003), $c = 4.80$ for the WT2G test collection, and $c = 5.58$ for the WT10G test collection. We use default values, instead of tuning the term weighting parameters, because our focus lies in testing our hypotheses, and not in optimising retrieval performance. If the said parameters are optimised, retrieval performance may be further improved. We measure the retrieval performance using the Mean Average Precision (MAP) measure (van Rijsbergen, 1979).

Throughout all experiments, we set *POS block* length at n = 4. We employ Good-Turing and Laplace smoothing, and set the threshold of high probability of occurrence empirically at θ = 0.01. We present all evaluation results in tables, the format of which is as follows: GT and LA indicate Good-Turing and Laplace respectively, and $\Delta\%$ denotes the % difference in MAP from the baseline. Statistically significant scores, as per the Wilcoxon test ($p < 0.05$), appear in boldface, while highest Δ percentages appear in italics.

4.2 Evaluation Results

Our retrieval baseline consists in testing the performance of each term weighting scheme, with each of the two test collections, using the original queries. We introduce two retrieval combinations on top of the baseline, which we call POS and POSC. The POS retrieval experiments, which relate to our first hypothesis, and the POSC retrieval experiments, which relate to our second hypothesis, are described in Section 4.2.1. Section 4.2.2 presents the assessment of our hypotheses using a performance-boosting retrieval technique, namely query expansion.

4.2.1 POS and POSC Retrieval Experiments

The aim of the POS and POSC experiments is to test our first and second hypotheses, respectively. Firstly, to test the first hypothesis, namely that there is a direct connection between the removal of low-frequency *POS blocks* from the queries and noise reduction in the queries, we remove all low-frequency *POS blocks* from the *narrative* field of

[4] http://snowball.tartarus.org/

[5] http://ir.dcs.gla.ac.uk/terrier/

the queries. Secondly, to test our second hypothesis as an extension of our first hypothesis, we refilter the queries used in the POS experiments by removing from them *POS blocks* that contain more closed class than open class tags. The processes involved in both hypotheses take place prior to the removal of stop words and stemming of the queries. Table 2 displays the relevant evaluation results.

Overall, the removal of low-probability *POS blocks* from the queries (*Hypothesis 1* section in Table 2) is associated with an improvement in retrieval performance over the baseline in most cases, which sometimes is statistically significant. This improvement is quite similar across the two statistical estimators. Moreover, two interesting patterns emerge. Firstly, the DFR weighting schemes seem to be divided, performance-wise, between the parametric BB2 and PL2, which are associated with the highest improvement in retrieval performance, and the non-parametric DLH, which is associated with the lowest improvement, or even deterioration in retrieval performance. This may indicate that the parameter used in BB2 and PL2 is not optimal, which would explain a low baseline, and thus a very high improvement over it. Secondly, when comparing the improvement in performance related to the WT2G and the WT10G test collections, we observe a more marked improvement in retrieval performance with WT2G than with WT10G.

The combination of our two hypotheses (*Hypotheses 1+2* section in Table 2) is associated with an improvement in retrieval performance over the baseline in most cases, which sometimes is statistically significant. This improvement is very similar across the two statistical estimators, namely Good-Turing and Laplace. When combining hypotheses 1+2, retrieval performance improves more than it did for hypothesis 1 only, for the WT2G test collection, which indicates that our second hypothesis might further reduce the amount of noise in the queries successfully. For the WT10G collection, we object similar results, with the exception of DLH. Generally, the improvement in performance associated to the WT2G test collection is more marked than the improvement associated to WT10G.

To recapitulate on the evaluation outcomes of our two hypotheses, we report an improvement in retrieval performance over the baseline for most, but not all cases, which is sometimes statistically significant. This may be indicative of successful noise reduction in the queries, as per our hypotheses. Also, the difference in the improvement in retrieval performance across the two test collections may suggest that data sparseness affects retrieval performance.

4.2.2 POS and POSC Retrieval Experiments with Query Expansion

Query expansion (QE) is a performance-boosting technique often used in IR, which consists in extracting the most relevant terms from the top retrieved documents, and in using these terms to expand the initial query. The expanded query is then used to retrieve documents anew. Query expansion has the distinct property of improving retrieval performance when queries do not contain noise, but harming retrieval performance when queries contain noise, furnishing us with a strong baseline, against which we can measure our hypotheses. We repeat the experiments described in Section 4.2.1 with query expansion.

We use the Bo1 query expansion scheme from the DFR framework (Amati, 2003). We optimise the query expansion settings, so as to maximise its performance. This provides us with an even stronger baseline, against which we can compare our proposed technique, which we tune empirically too through the tuning of the threshold θ. We optimise query expansion on the basis of the corresponding relevance assessments available for the queries and collections employed, by selecting the most relevant terms from the top retrieved documents. For the WT2G test collection, the relevant terms / top retrieved documents ratio we use is (i) 20/5 with TF_IDF, BM25, and DLH; (ii) 30/5 with PL2; and (iii) 10/5 with BB2. For the WT10G collection, the said ratio is (i) 10/5 for TF_IDF; (ii) 20/5 for BM25 and DLH; and (iii) 5/5 for PL2 and BB2.

We repeat our POS and POSC retrieval experiments with query expansion. Table 3 displays the relevant evaluation results.

Query expansion has overall improved retrieval performance (compare Tables 2 and 3), for both test collections, with two exceptions, where query expansion has made no difference at all, namely for BB2 and PL2, with the WT10G collection. The removal of low-probability *POS blocks* from the queries, as per our first hypothesis, combined with query expansion, is associated with an im-

Table 2: Mean Average Precision (MAP) scores of the POS and POSC experiments.

WT2G collection									
		Hypothesis 1				Hypotheses 1+2			
w(t,d)	base	POSGT	Δ%	POSLA	Δ%	POSCGT	Δ%	POSCLA	Δ%
TFIDF	0.276	0.295	+6.8	0.293	+6.1	0.298	+8.0	0.294	+6.4
BM25	0.280	0.294	+4.8	0.292	+4.1	0.297	+5.9	0.293	+4.5
BB2	0.237	0.291	***+22.8***	0.287	*+21.0*	0.295	***+24.2***	0.288	*+21.5*
PL2	0.268	0.298	+11.2	0.297	+10.9	0.306	+14.1	0.302	+12.8
DLH	0.237	0.239	+0.7	0.238	+0.4	0.243	+2.3	0.241	+1.6
WT10G collection									
		Hypothesis 1				Hypotheses 1+2			
w(t,d)	base	POSGT	Δ%	POSLA	Δ%	POSCGT	Δ%	POSCLA	Δ%
TFIDF	0.231	0.234	+1.2	0.238	+2.8	0.233	+0.7	0.237	+2.6
BM25	0.234	0.234	none	0.238	+1.5	0.233	-0.4	0.237	+1.2
BB2	0.206	0.213	+3.5	0.214	+4.0	0.216	+5.0	0.220	+6.7
PL2	0.237	0.253	*+6.8*	0.253	***+7.0***	0.251	*+6.1*	0.256	***+8.2***
DLH	0.232	0.231	-0.7	0.233	+0.5	0.230	-1.0	0.234	+0.9

Table 3: Mean Average Precision (MAP) scores of the POS and POSC experiments with Query Expansion.

WT2G collection									
		Hypothesis 1				Hypotheses 1+2			
w(t,d)	base	POSGT	Δ%	POSLA	Δ%	POSCGT	Δ%	POSCLA	Δ%
TFIDF	0.299	0.323	+8.0	0.329	+10.0	0.322	+7.7	0.325	+8.7
BM25	0.302	0.320	+5.7	0.326	+7.9	0.319	+5.6	0.322	+6.6
BB2	0.239	0.291	***+21.7***	0.288	*+20.5*	0.291	***+21.7***	0.287	*+20.1*
PL2	0.285	0.312	+9.5	0.315	+10.5	0.315	+10.5	0.316	+10.9
DLH	0.267	0.283	+6.0	0.283	+6.0	0.284	+6.4	0.283	+6.0
WT10G collection									
		Hypothesis 1				Hypotheses 1+2			
w(t,d)	base	POSGTQE	Δ%	POSLAQE	Δ%	POSCGT	Δ%	POSCLA	Δ%
TFIDF	0.233	0.241	+3.4	0.249	***+6.9***	0.240	+3.0	0.250	+7.3
BM25	0.240	0.248	+3.3	0.250	+4.2	0.244	+1.7	0.249	+3.7
BB2	0.206	0.213	+3.4	0.214	+3.9	0.216	+4.8	0.220	+6.8
PL2	0.237	0.253	*+6.7*	0.253	+6.7	0.251	*+5.9*	0.256	***+8.0***
DLH	0.236	0.250	+5.9	0.246	+4.2	0.250	*+5.9*	0.253	+7.2

provement in retrieval performance over the new baseline at all times, which is sometimes statistically significant. This may indicate that noise has been further reduced in the queries. Also, the two statistical estimators lead to similar improvements in retrieval performance. When we compare these results to the ones reported with identical settings but without query expansion (Table 2), we observe the following. Firstly, the previously reported division in the DFR weighting schemes, where BB2 and PL2 improved the most from our hypothesised noise reduction in the queries, while DLH improved the least, is no longer valid. The improvement in retrieval performance now associated to DLH is similar to the improvement associated with the other weighting schemes. Secondly, the difference in the retrieval improvement previously observed between the two test collections is now smaller.

To recapitulate on the evaluation outcomes of our two hypotheses combined with query expansion, we report an improvement in retrieval performance over the baseline at all times, which is sometimes statistically significant. It appears that the combination of our hypotheses with query expansion tones down previously reported sharp differences in retrieval improvements over the baseline (Table 2), which may be indicative of further noise reduction.

5 Conclusion

We described a block-based part of speech (POS) modeling of language distribution, induced from a corpus, and statistically smoothened using two different estimators. We hypothesised that high-frequency *POS blocks* bear more content than low-frequency *POS blocks*. Also, we hypothesised that the more closed class components a *POS block* contains, the less content it bears. We evaluated both hypotheses in the context of Information Retrieval, across two standard test collections, and five statistically different term weighting schemes. Our hypotheses led to a general improvement in retrieval performance. This improvement was overall higher for the smaller of the two collections, indicating that data sparseness may have an effect on retrieval. The use of query expansion worked well with our hypotheses, by helping weaker weighting schemes to benefit more from the reduction of noise in the queries.

In the future, we wish to investigate varying the size n of *POS blocks*, as well as testing our hypotheses on shorter queries.

References

Alan F. Smeaton. 1999. *Using NLP or NLP resources for information retrieval tasks. Natural language information retrieval*. Kluwer Academic Publishers Dordrecht, NL.

Bruce Croft and John Lafferty. 2003. *Language Modeling for Information Retrieval*. Springer.

Christopher D. Manning and Hinrich Schutze. 1999. *Foundations of Statistical Language Processing*. The MIT Press, London.

David D. Lewis. 1992. An Evaluation of Phrasal and Clustered Representations on a Text Categorization Task. *ACM SIGIR 1992*, 37–50.

Gianni Amati. 2003. *Probabilistic Models for Information Retrieval based on Divergence from Randomness*. Ph.D. Thesis, University of Glasgow.

Ingrid Zukerman and Bhavani Raskutti. 2002. Lexical Query Paraphrasing for Document Retrieval. *COLING 2002*, 1177–1183.

John Lyons. 1977. *Semantics: Volume 2*. CUP, Cambridge.

Karen Sparck-Jones. 1972. A statistical interpretation of term specifi city and its application in retrieval. *Journal of Documentation*, 28:11–21.

'Keith' (C. J.) van Rijsbergen. 1979. *Information Retrieval*. Butterworths, London.

Kenneth W. Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*, 19:313–330.

Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jennifer C. Lai, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Stephen Robertson, Steve Walker, Micheline Beaulieu, Mike Gatford, and A. Payne. 1995. Okapi at TREC-4. *NIST Special Publication 500-236: TREC-4*, 73–96.

Tomek Strzalkowski. 1996. Robust Natural Language Processing and user-guided concept discovery for Information retrieval, extraction and summarization. *Tipster Text Phase III Kickoff Workshop*.