PART OF SPEECH N-GRAMS AND INFORMATION RETRIEVAL

Christina Lioma et C.J. Keith van Rijsbergen

Pub. linguistiques | *Revue française de linguistique appliquée*

2008/1 - Vol. XIII
pages 9 à 22

ISSN 1386-1204

Rev. franç. de linguistique appliquée, 2008, XIII-1 (9-22)

Part of speech n-grams and Information Retrieval

Christina Lioma, Katholieke Universiteit Leuven[1]
C.J. Keith van Rijsbergen, University of Glasgow

Abstract: *Efforts to use linguistics in information retrieval (IR) were initiated in the 1980s, and intensified in the 1990s, reporting performance benefits (see the overviews by Smeaton 1986 & 1999, Karlgren 1993, and Tait 2005). After that time, these efforts decreased: baseline system performance improved, and the cost associated with linguistic processing was not worth the small benefits over the already improved baselines (Tait, 2005). At present, most research on linguistics for IR tends to be geared towards domain-specific IR applications that seem to benefit more from linguistics, like question-answering (Tait & Oakes 2006). Although such applications are important, they should not limit the scope of research into linguistics for IR. In this work, we present an alternative use of linguistics, part of speech information in particular, to compute a term weight of informative content. This term weight is a novel application of linguistics to IR, and can benefit retrieval performance of general IR systems.*

Résumé : *Les tentatives d'utilisation de connaissances linguistiques en recherche d'information (RI) ont commencé dans les années 1980 et se sont développées dans les années 1990, en mettant en évidence des améliorations de performance (voir les synthèses de Smeaton 1986 et 1999, Karlgren 1993, et Tait 2005). Depuis lors, ces tentatives sont allées décroissant : les performances des systèmes basiques se sont améliorées et le coût du traitement linguistique ne justifiait pas le petit bénéfice obtenu (Tait 2005). La plupart des recherches en linguistique pour la RI ont aujourd'hui tendance à se tourner vers les applications de domaines spécifiques, qui semblent mieux bénéficier de ces connaissances, comme les systèmes de question-réponse (Tait & Oakes 2006). Bien que ces applications soient importantes, elles ne couvrent pas toute la recherche en linguistique pour la RI. Dans cet article, nous présentons une autre utilisation de la linguistique, plus précisément des informations sur les catégories grammaticales, pour pondérer le contenu informatif de séquences de texte. Cette pondération est une nouvelle application de la linguistique en RI et peut améliorer la performance des systèmes en général.*

1. Introduction

Information Retrieval (IR) systems aim to locate and quantify information in data with respect to some user query. A common example of IR systems is World Wide Web (Web) search engines, in which a short keyword query is used to generate a ranked list from a pre-indexed heterogeneous collection of documents. The matching between queries and documents is mostly term-based, i.e. the words within documents are used to describe the documents and to determine their relevance to a given query. Even though this type of statistical modelling of

[1] Work realised at the University of Glasgow. Presently, the author is affiliated to Katholieke Universiteit Leuven.

documents generally lacks transparent knowledge of language, numerous such techniques have become standard in the field, e.g. the Okapi model for term weighting and document ranking (Robertson & Walker 1994). IR techniques using such statistical models almost always outperform more linguistically based ones (Tait & Oakes 2006). For example, most users of Google find enough relevant documents in the first page without any linguistic sophistication. Recently computational linguistics have made significant contributions to specialised areas of IR, for instance question answering (Tait & Oakes 2006). However, very little linguistics is used in the development of mainstream IR systems. In this work, we propose a novel application of linguistics in IR, which can be used in general IR systems.

We propose to use part of speech (POS) information in IR in order to compute a *term weight*. Term weights are mathematical computations of how informative words are, and constitute an integral part of the statistical modelling of documents by IR systems. Broadly speaking, term weights aim to reward discriminative words, which in the context of IR are defined as words occurring frequently in a document but not so frequently in a general collection of documents (Spärck Jones 1972). We propose an alternative term weight, which is computed not from word frequency statistics, but from POS statistics: our term weight rewards terms occurring often in content-rich contexts, e.g. often co-occurring with nouns, verbs or adjectives. The goal of our proposed term weight is to consider the shallow grammatical information of terms (represented by their POS) and the context in which they occur. Experimental evaluation confirms that our term weight can benefit retrieval.

This paper is organised as follows. Section 2 discusses the motivation for deriving a term weight from POS information. Section 3 presents our notation. Section 4 presents how we derive a term information score from POS. Section 5 presents the experimental evaluation of our proposed term weight as part of an IR system. Section 6 concludes this work.

2. Motivation for computing a term weight from parts of speech

We propose a term weight that is computed from POS information, motivated by the fact that POS can indicate to an extent the presence or absence of informative content in language. This is not new. Early POS categorisations date back to 4th century BC studies of Sanskrit and ancient Greek, which split grammatical categories of words roughly into three *classes* (Lyons 1977):

(i) subject of a predication → **nouns**
(ii) action or quality predicated → **verbs**[2] & **adjectives**
(iii) peripheral/functional use → everything else

A more recent formulation of this POS distinction is Jespersen's *Rank Theory* (Jespersen, 1913, 1929), which suggested that grammatical categories are semantically definable and subject to ranking. Jespersen identified POS *degrees*:

(i) *1st degree* (or *primary*) POS → **nouns**
(ii) *2nd degree* (or *secondary*) POS → **verbs**[2] & **adjectives**
(iii) *3rd degree* (or *tertiary*) POS → **adverbs**

Jespersen defined the notion of *degree* in terms of the POS combinatorial properties: each POS is modified by a POS of higher degree. E.g. nouns are modified by verbs, and verbs are modified by adverbs. No more than three degrees are required, because there is no major POS with the function to modify POS of the 3rd degree.

[2] Including participles.

A more general POS distinction is between *open* and *closed* class POS:

(i) open POS (= dynamic vocabulary, mainly content-bearing) → **nouns**, **verbs**, **adjectives** (Table 1, in bold)
(ii) closed POS (= controlled vocabulary, mostly functional) → everything else

The open-closed class POS distinction is widely accepted. Linguists often compare it to the Aristotelian opposition of 'matter' and 'form': open class POS 'signify' the objects of thought which constitute the 'matter' of discourse; closed class POS do not 'signify' much of themselves, but instead contribute to the total meaning of sentences, by imposing upon them a certain 'form' or organisation (Bas & *al.* 2004, 29-64). Practitioners of language processing technologies often consider closed class POS words as 'stopwords' and exclude them from processing, because of their negligible contribution to the overall content of the text being processed. The open-closed class POS separation is also reminiscent of the distinction traditionally drawn between 'full' and 'empty' words in Chinese grammatical theory (Lyons 1977, 273).

Penn Treebank classification: primary parts of speech			
Part of speech	***Abbr.***	***Part of speech***	***Abbr.***
Adjective	**JJ**	**Participle**	**VR**
Adverb	RB	Particle	RP
Conjunction	CC	Possessive ending	PO
Determiner	DT	Preposition	IN
Modal verb	MD	Pronoun	PP
Noun	**NN**	Symbol	SY
Numeral	CD	**Verb**	**VB**

Table 1. *Primary part of speech (POS) categories of the Penn Treebank set* (Marcus & *al.* 1993). Abbr. = abbreviations. Open class POS are printed in bold.

Generally, we use POS, as opposed to other linguistic information like semantic or discourse structure, because:

- we wish to capture non-topical information about words and use it in IR. Non-topical information refers to how informative a word is in general, not how informative word A is with respect to word B. POS information is better suited for this task, e.g. we can assume that nouns are generally informative, without having semantic or discourse knowledge about their use.
- POS are a small and finite set of categories, hence better suited for the IR task: text can be easily/quickly annotated using existing POS tagging technology, with relatively high expected accuracy (state of the art POS tagging performance approaches > 90% accuracy).

We use POS information in the form of n-grams (POS n-grams), which are contiguous POS sequences (e.g. determiner-adjective-noun, adjective-noun-verb, noun-verb-adverb, and so on). We use POS n-grams because they encode two types of information (Figure 1):

(i) POS → shallow grammatical information, which can indicate to an extent the presence/absence of content;
(ii) N-grams → 'small windows', which can represent contextual information.

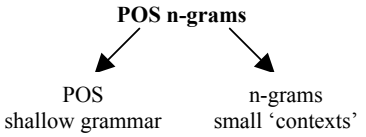

Figure 1. *POS n-grams encode both shallow grammatical and contextual information.*

Overall, we consider POS n-grams as 'POS contexts', for which we can assume some prior knowledge of content, e.g. POS n-grams containing nouns and verbs are likely to be more informative than POS n-grams containing prepositions and adverbs. Our intuition for deriving a term weight is that a term that occurs in many term n-grams, which themselves correspond to informative POS n-grams, is likely to be informative. We look at all the POS n-grams of a term, and we reason that the more informative and frequent these POS n-grams are, the more informative that term is likely to be.

3. Part of speech n-gram notation

Given a contiguous sequence of items, an n-gram is a contiguous subsequence of that sequence. Following the n-gram notation of Brown & *al*. (1992), let i_h^k be a contiguous sequence of items, where i_h is the first item, and i_k is the last item in the sequence. Then, $i_{j=1}^n$ is a contiguous subsequence or string of that sequence, if n does not exceed the length of the sequence ($i_j \geq h$ and $n > k$). Such subsequences are called n-grams, usually when the number of items n in the string is fixed, and when they are extracted in a recurrent and overlapping way from the initial sequence (Damerau 1971). For example, given the sequence *the cat sat on the mat*, and for n=3:

$$t_1^3 = \textit{the cat sat}$$
$$t_2^4 = \textit{cat sat on}$$
$$t_3^5 = \textit{sat on the}$$
$$t_4^6 = \textit{on the mat}$$

The total of all sequences from which n-grams are extracted is often called *sample*. n-grams are usually extracted from very large samples. Usually, the likelihood of observing an n-gram in the sample is assigned a probability, so that the more frequent the n-gram is in the sample, the higher its probability of occurrence. The computational mechanism for obtaining these probabilities is referred to as a *language model* (Brown & *al.* 1992). (See Feller (1950) and Gallager (1968) for more.) Hence, a language model is a probability distribution over sets of n-grams. The value of n is called the *order* of the language model, and controls the amount of context captured inside the n-gram.

Let $i_{j=1}^n$ be a term n-gram, $pos_{j=1}^n$ be a POS n-gram, and ϕ' be a function that maps terms to POS (i.e. POS tagging):

$$\phi'(t_{j=1}^n) = pos_{j=1}^n \quad (1)$$

Applying function (1) to the above example gives the following POS n-grams[3]:

ϕ' *(the cat sat) = DT NN VB*

ϕ' *(cat sat on) = NN VB IN*

ϕ' *(sat on the)*

ϕ' *(on the mat) = IN DT NN*

In order to compute a term weight from POS n-grams, we need to make the link between terms and POS. We do this by relating a term to all the POS n-grams that 'contain' it (= POS n-grams corresponding to term n-grams containing this term). Let $\{t_{j=1}^{n}\}_i$ be the set of all term n-grams $\{t_{j=1}^{n}\}$ that contain term t_i *at any position inside the n-gram*. For example, given the above 3-grams:

for t_i *= cat,*

$\{t_{j=1}^{n}\}_i$ *= {the cat sat, cat sat on}*$_{cat,}$

for t_i *= the,*

$\{t_{j=1}^{n}\}_i$ *= {the cat sat, sat on the, on the mat}*$_{the.}$

Then,

$$\phi''(\{t_{j=1}^{n}\}_i) = \{pos_{j=1}^{n}\}_i \quad (2)$$

where φ'' denotes applying function (1) to each term n-gram in the set $\{t_{j=1}^{n}\}_i$. E.g.

φ'' *({the cat sat, cat sat on}*$_{cat}$*) = {DT NN VB, NN VB IN}*$_{cat,}$

φ'' *({the cat sat, sat on the, on the mat}*$_{the}$*) = {DT NN VB, VB IN DT, IN DT NN}*$_{the.}$

The relation described by function 2 is important for computing a term weight from POS n-grams, because it maps a term to all the POS n-grams that 'contain' it. How this is used to compute a term weight is shown next.

4. Part of speech information score

For a term, we compute a term weight, which we call *POS Information Score* (PIS), on the basis of how informative this term is and how informative are the terms it generally co-occurs with. We use two types of information: (i) POS class and degree (= *a priori* information about how informative a term is in general); (ii) POS n-gram statistics (= observed information about how terms co-occur). We combine these two types of information using basic probability principles (see Good 1968).

There are 3 steps in computing PIS (summarised Table 2):

Step 1 compute the probability that an individual POS is informative, using POS class and degree information (described in Section 4.1);

Step 2 extend Step 1 to compute the probability that a POS n-gram is informative (described in Section 4.2);

[3] Part of speech abbreviations are presented in Table 1.

Step 3 extend Step 2 to compute the probability that an individual term is informative, by mapping term n-grams containing this term to their corresponding POS n-grams (described in Section 4.3).

Methodology for computing PIS	
Step 1	Probability that an individual part of speech is informative (P(inf|pos))
Step 2	Probability that a POS n-gram is informative (P(inf|pos n-gram))
Step 3a	For a term, get all the term n-grams that contain it
Step 3b	Map all these term n-grams to their corresponding POS n-grams
Step 3c	The total P(inf|pos n-gram) of these POS n-grams gives PIS

Table 2. *Step by step methodology for computing the part of speech information score (PIS) for a term*.

4.1. Probability that an individual part of speech is informative (Step 1)

Using probability notation, $P(x)$ denotes the probability that event x occurs, ($0 \leq P(x) \leq 1$). For our computation, let inf be the event of informative content, and let pos be the event of a POS. Then, $P(\text{inf} \mid pos)$ is the conditional probability that inf occurs given pos, or more simply the probability of pos being informative. Then, the distinction between open and closed POS can be written as:

$$0 \leq P(\text{inf} \mid pos_o) \leq 1 \quad (3)$$

$$P(\text{inf} \mid pos_c) = 0 \quad (4)$$

where pos_o and pos_c is an open and closed class POS, respectively. Equation 3 states that there is always some probability of an open class POS being informative. Equation 4 states that there is no probability of a closed class POS being informative. These are assumptions that do not always hold: a closed class POS can be informative. These assumptions are made as approximations.

Open class POS can be either 1st or 2nd degree POS, according to Jespersen's Rank Theory. For 1st degree POS, Equation 3 can be modified, on the assumption that a 1st degree POS is always informative:

$$P(\text{inf} \mid pos') = \lambda, \ (0 < \lambda \leq 1) \quad (5)$$

where pos' is a 1st degree POS. Equation 5 states that the probability of nouns being informative is λ. For 2nd degree POS, Equation 4 can be modified, on the assumption that 2nd degree POS are always informative, but less than 1st degree POS:

$$P(\text{inf} \mid pos'') = \rho, \ (0 < \rho < \lambda) \quad (6)$$

where pos'' is a 2nd degree POS. Equation 6 states that the probability of verbs, participles, and adjectives being informative is ρ.

We suggest two alternatives for computing λ and ρ, one empirical and one formal: Firstly, λ and ρ can be tuned empirically to optimise the performance of a process that uses POS. For example, in this work, we use POS n-grams as part of an IR system. Hence, λ and ρ can be tuned to maximise the performance (e.g., Mean Average Precision (MAP)) of the IR system:

$$\arg\max_{\lambda|\rho} = MAP(\lambda \mid \rho) \qquad (7)$$

Secondly, a value for λ and ρ can be derived formally from the probabilities of individual POS being informative, by using Bayes rule for combining probabilities (see Appendix I).

4.2. Probability that a part of speech n-gram is informative (Step 2)

We estimate how informative a POS n-gram is, on the basis of how informative its members are. (The members of a POS n-gram are individual POS.) For each member of a POS n-gram, we already have a probability of how informative it is from Step 1. The combination of these probabilities is an approximation of how informative the POS n-gram is.

Let $pos_{j=1}^{n}$ be a POS n-gram. Then, we compute the probability that $pos_{j=1}^{n}$ is informative $P(\text{inf} \mid pos_{j=1}^{n})$ as the sum of the probabilities of each of its members being informative:

$$P(\text{inf} \mid pos_{j=1}^{n}) = \sum_{j=1}^{n} P(\text{inf} \mid pos) \cdot P(pos) \qquad (8)$$

where $P(\text{inf} \mid pos)$ is the probability that an individual part of speech is informative, computed using Equations 3 - 6, and $P(pos)$ is the probability of a POS occurring in the POS n-gram: $P(pos) = 1/n$.

Closed class POS can be excluded from Equation 8, because they are assumed to be non-informative always (Equation 4). Equation 8 needs to process only open class POS. Hence, to compute the probability that a POS n-gram is informative, we need to know n, λ and ρ (for 2nd degree POS). n is known a priori. λ and ρ can be computed using either of the two ways suggested above.

In Equation 8, an alternative way to the linear combination of the probabilities $P(\text{inf} \mid pos)$ inside a POS n-gram would be to compute their product or sum their logarithms. Generally, these alternatives are considered approximately equivalent. We choose the linear combination for simplicity. (Summation of probabilities is simpler, because multiplication would require smoothing[4], and summation of logarithms would be computationally costly[5].)

4.3. Probability that a term is informative (Step 3)

So far, we have computed the probability that a POS n-gram is informative. We now present how to compute the probability that a term is informative, which is our term weight (PIS). For a term, we estimate PIS by doing two things: 1) We map all the term n-grams in which the term occurs to their corresponding POS n-grams; 2) We combine the probabilities that each of these POS n-grams is informative.

[4] Without smoothing, a zero probability nullifies the product, and a probability of 1 does not contribute to the product.
[5] Computing logarithms is considered a computationally expensive process.

The set of all POS n-grams which correspond to a term n-gram containing term t_i was defined in Section 3 as $\{pos^n_{j=1}\}_i$. Using this, the probability of term t_i being informative is:

$$P(\text{inf} \mid t_i) = \sum_{j=1}^{n} P(\text{inf} \mid \{pos\}_i) \cdot P(pos), \qquad (0 \leq P(\text{inf} \mid t_i \leq 1)) \quad (9)$$

$P(\text{inf} \mid \{pos\}_i)$ is computed with Equation 8 by replacing $\{pos\}$ with $\{pos\}_i$, and $P(pos)$ is the probability of a POS n-gram occurring: $P(pos) = 1/|C|$, where $|C|$ is the number of all POS n-grams in the collection.

Equation 9 states that the probability of a term being informative is a function of how informative are all the POS contexts in which it occurs. The reasoning is that a term that occurs in many term n-grams, which themselves correspond to informative POS n-grams, is likely to be informative. This is quantified by combining the informative content of these POS n-grams, and their probability of occurrence. More simply, PIS is the ratio of how informative all POS n-grams 'containing' a term are, over how many POS n-grams occur in the collection.

The use of POS n-grams to compute a term weight, shown in Equation 9, begs the question: Why compute a term weight from POS n-grams, and not simply from individual POS? By using single parts of speech, instead of POS n-grams, it would be possible to compute a term information score only for terms of open class POS, and not for terms of closed class POS. Also, and most importantly, this score would not model the 'POS context' in which terms occur, but it would only be a simple function of the total number of terms in the collection. On the contrary, by using POS n-grams instead of individual POS to compute PIS, we look at all the POS n-grams 'containing' a term in a collection, and hence we model all the 'POS contexts' in which a term occurs. These 'POS contexts' contribute to PIS the following: how informative the terms co-occurring with a given term are. The more informative these co-occurring terms, the higher the value of PIS. Also, the more often such terms co-occur, the higher the value of PIS. Hence, PIS is not restricted to terms of open class POS only, and also, it does not correspond to a 'flat' score for all terms of the same POS. Table 3 illustrates this point by showing the PIS values of sample terms[6]. These term weights have been computed using Equation 9, with POS 4-grams extracted from the WT10G collection, which is presented in Section 4. The term frequency in the collection is also presented for comparison with PIS. We expect lower frequencies to indicate more discriminative words, i.e. higher term weights.

Table 3 shows that, overall, term frequency in the collection and PIS tend to agree, however, they are not identical. For instance, several terms of similar frequency in the collection and/or of the same POS have different PIS. For example: *recall* - *tours*: same frequency (2), same part of speech (noun[7]), different PIS (0.3343 - 0.4388); *mary* - *lady*: similar frequency (143 - 164), same part of speech (noun[8]), different PIS (0.5005 - 0.3937); *jose* - *dental* - *symptom*: similar frequency (35,629 - 36,821 - 36,881), different part of speech (noun - adjective - noun), different PIS (0.4163 - 0.2886 - 0.3453). Also, there exist terms of similar PIS, but of different frequency and/or POS. For instance, *you* - *world*: similar PIS

[6] These terms are taken from the top 500 search engine keywords of the wordtracker Web site: http://www.searchengineguide.com/wt/2007/0822_wt1.html.
[7] Either of these terms can also be a verb.
[8] To be precise, *mary* is a proper noun. In this work, we do not distinguish between different noun classes, as described in Section 2.1.

(0.2848 - 0.2891), different frequency (214 - 773,497), different POS (pronoun - noun). These examples illustrate the point that the PIS of a term is not simply a function of its frequency in the collection and how informative its POS is.

Term	Frequency	PIS	Term	Frequency	PIS
recall	2	0.3343	symptom	36,881	0.3453
Tours	2	0.4388	anderson	38,907	0.3884
facebook	97	0.4114	aol	129,000	0.3292
Mary	143	0.5005	yahoo	138,589	0.3645
Lady	164	0.3937	hot	138,796	0.2402
You	214	0.2848	weather	155,278	0.3032
Paris	276	0.4885	radio	156,908	0.3315
mattel	672	0.4582	station	162,711	0.3217
walmart	684	0.4312	english	177,158	0.2645
halen	1,201	0.5173	white	234,691	0.2509
jameson	1,549	0.4261	video	251,345	0.2913
hotmail	1,684	0.4600	west	219,494	0.2686
sonia	1,743	0.4936	park	281,315	0.2828
Play	2,484	0.2829	job	343,179	0.2828
Nile	5,175	0.4136	game	359,590	0.3109
nigeria	5,475	0.3982	care	383,359	0.3109
Porn	5,744	0.4013	music	411,467	0.2710
hilton	7,563	0.4365	local	467,758	0.2136
cheat	8,414	0.2598	free	523,669	0.2050
pamela	8,761	0.4470	find	524,453	0.1990
gospel	14,946	0.3720	world	773,497	0.2891
Tube	31,383	0.3496	mail	855,685	0.2423
Jose	35,629	0.4163	Name	901,525	0.2275
dental	36,821	0.2886	Home	1,365,190	0.2567

Table 3. *Example: terms, their frequency in the WT10G collection, and their part of speech term weight (PIS).*
PIS is computed with Equation 9, using POS 4-grams from WT10G.

5. Experimental evaluation

So far, we have derived a term weight from POS n-grams, called PIS. Here, we suggest how PIS can enhance the IR process. The hypothesis is that using PIS can improve retrieval performance, because when computing how informative document terms are with respect to query terms, the non-topical information (given by PIS) can 'boost' the relevance score of generally informative terms, and decrease the score of generally non-informative terms. Section 5.1 presents the experimental methodology and settings. Section 5.2 presents the experimental results.

5.1. Experimental methodology and settings

The experimental setting is an IR system that matches documents to queries using an established retrieval model (baseline). To test the hypothesis, we integrate PIS into the model, and compare its performance to that of the baseline. We realise two rounds of experiments: 1) with default settings, and 2) with settings optimised for retrieval performance. The aim is to show that PIS can improve retrieval performance first on standard settings, and then on a stronger baseline.

We integrate PIS into the retrieval model by multiplying it to the relevance score of a document for a query term:

$$rel(d,q) = \sum_{t \in q} w(t,d) \cdot PIS_t \quad (10)$$

where $rel(d,q)$ is the relevance score between a document and a query, and $w(t,d)$ is the weight of relevance between a query term and a document, computed by the retrieval model. This type of integration of PIS into the relevance score by simple multiplication resembles the way in which prior probabilities of relevance, or *priors,* are sometimes integrated into the retrieval process (Kraaij & *al*. 2002). Even though their integration can be similar, there is a fundamental difference between PIS and such priors: Typically, these priors represent the likelihood of a document being relevant, i.e. they apply to documents and they are heuristically-driven. For instance, based on the observation that longer documents are more often relevant to queries, it has been assumed that the probability of a document being relevant to any query is higher for longer documents (Blanco & Barreiro 2008). Our proposed term weight is applied to individual terms, not documents, and it is based on a well-known POS categorisation, hence it represents the likelihood of a term being informative.

We retrieve documents from the WT10G and Disks 4&5 TREC[9] collections, using their corresponding queries: queries 451-550 for WT10G, and queries 301-450 & 601-700 for Disks 4&5. TREC queries usually contain a *title*, *description*, and *narrative* portion. The title contains few keywords; the description includes a brief description of the information need; the narrative contains a longer description of the information need. We experiment separately with short queries (title portion) and long queries (description portion). We evaluate retrieval performance in terms of Mean Average Precision (MAP) and Precision at 10 (P10), and report the results of statistical significance tests, using the Wilcoxon matched-pairs signed-ranks test.

We conduct experiments using the Terrier IR system (Ounis & *al.* 2007). Before retrieval, terms are tokenised on whitespace and punctuation marks, and lower-cased; stopwords are removed and terms are stemmed with the Porter stemmer (Porter 1980). We match documents to queries with the Okapi Best Match 25 (BM25) model (Robertson & Walker 1994). BM25 includes a tunable parameter b[10], which has a 'smoothing' role, *i.e*. it 'normalises' the relevance score of a document for a query across document lengths, in order to avoid bias towards longer documents. First, we use the default setting (b=0.75); then, we optimise b for retrieval performance, by varying it within (0,1] with a unique interval of 0.05. The value that gives the best retrieval performance is considered optimal. We optimise separately for MAP and P10, for different query lengths and collections.

We compute PIS using POS n-grams from the same collections used for retrieval. To do so, collections are POS tagged with the TreeTagger (Schmid 1997), and POS 4-grams are extracted[11]. PIS includes variables λ and ρ[12], which correspond to the probability that a 1st and 2nd degree part of speech is informative, respectively. We set λ=1, i.e. we assume that nouns are always informative. Then, first, we formally derive a default value for ρ using

[9] Text REtrieval Conference (TREC): http://trec.nist.gov/.

[10] BM25 also includes parameters k_1 and k_3, which do not have a noted impact on retrieval performance. We use their recommended values: $k_1 = 1.2$ and $k_3 = 1000$ (Robertson & Walker 1994).

[11] We can report that ranging n-gram values between n=1 - 100 shows that n=4 - 6 gives similar results. These experiments appear in Lioma (2008).

[12] λ and ρ are probabilities, not parameters. We treat them as variables, and tune them to retrieval performance.

Bayes theory (see Appendix I), which gives $\rho=0.33$. Second, we optimise ρ for retrieval performance by varying it within (0,1] with a unique interval of 0.05. We select the value that gives the best MAP and P10, separately for different query lengths and collections.

5.2. Experimental results

Table 4 shows the results of our experiments. The best scores for each row are printed in bold. The asterisk * shows strong statistical significance (at < 0.01**)**. We see that integrating PIS into the retrieval process either improves or does not alter retrieval performance, but never harms it. In particular, long queries benefit more from PIS than short queries (between +3.1% and +14.3% for long queries, between none and +3.3% for short queries). This is not surprising: longer queries contain more words, which are not necessarily the most relevant content-bearing words (*keywords*). Short queries tend to contain few words, which are mainly keywords. Keywords are likely to be informative, hence the contribution of PIS is small. Compared to short queries, long queries tend to contain more terms, which are not necessarily informative, hence the contribution of PIS is bigger.

Eval.	**Short queries**				**Coll.**
	Def. BM25	**Def. BM25 + PIS**	**Opt. BM25**	**Opt. BM25 + PIS**	
MAP	0.242	0.242 (none)	**0.254**	**0.254** (none)	Disks4&5
P10	0.424	0.424 (none)	0.438	**0.442** (+0.9%)	
MAP	0.187	0.188 (+0.5%)	0.211	**0.212** (+0.4%)	WT10G
P10	0.300	0.310 (+3.3%)	0.328	**0.337** (+2.7%)	
Eval.	**Long queries**				**Coll.**
	Def. BM25	**Def. BM25 + PIS**	**Opt. BM25**	**Opt. BM25 + PIS**	
MAP	0.242	0.256 (+5.8%)*	0.244	**0.259** (+6.1%)*	Disks4&5
P10	0.423	0.436 (+3.1%)*	0.423	**0.437** (+3.3%)*	
MAP	0.175	0.200 (+14.3%)*	0.187	**0.211** (+12.8%)*	WT10G
P10	0.334	0.356 (+6.6%)*	0.344	**0.362** (+5.2%)*	

Table 4. *Retrieval with the BM25 retrieval model from the WT10G and Disks 4&5 collections with short and long queries*.

Mean average precision (MAP) and precision at 10 (P10) are shown for default (def.) and optimised (opt.) parameters. (Parameters are optimised for MAP and P10 separately.) PIS = our POS n-gram based term weight. (%) = % difference in MAP or P10 from the baseline. * = stat. significance at <0.01 with the Wilcoxon matched-pairs signed-ranks test. Bold = best scores per row. Overall, PIS is associated with improved retrieval performance, especially for longer queries.

We also note that when integrating PIS into retrieval and measuring performance, the benefit is more for MAP than for P10. One reason for this could be that using PIS alters the relevance ranking of documents with respect to a query, less at the top ranks (measured by P10), and more at the lower ranks (measured by MAP). Hence, this could indicate that PIS benefits recall (how many documents are retrieved) slightly more than it benefits precision (how relevant are the documents retrieved). More experiments are needed to test whether this is true.

Finally, we note that retrieval performance with PIS into the model is better for WT10G than it is for Disks 4&5. This could be due to the fact that the baseline model (without PIS) gave lower scores for WT10G than it did for Disks 4&5, at all times. Hence, since the

baseline performed worse on WT10G, there was more room for improvement on this collection, than there was in Disks 4&5. This improvement was made by using PIS.

The above observations are true for both collections, query lengths, evaluation measures, and settings used, hence they are solid indications that our proposed term weight, computed from POS n-grams, can enhance retrieval performance.

In this section, we tested the hypothesis that our proposed non-topical term information score, which is derived from POS n-grams, can be combined with the topical information of terms, computed with conventional retrieval models, to improve retrieval performance. We suggested a simple integration of PIS into the retrieval model. A series of retrieval experiments using both default and optimal settings showed that PIS can improve retrieval performance.

6. Conclusion

We investigated an application of linguistics to the standard information retrieval (IR) process. Specifically, we proposed an application of part of speech (POS) n-grams to compute a term weight that represents how informative terms are in general. Our main argument is that POS n-grams encode grammatical and structural information about language in a shallow way, which can be statistically manipulated to indicate the non-topical informative content of words. Non-topical content refers to how informative a word is in general, and not with respect to a topic. We addressed two main issues in this work. First, we introduced a framework for representing parts of speech as contiguous sequences (n-grams), and, within this framework, we used basic principles of probability theory to derive a non-topical information score for words. Second, we evaluated this term information score as part of the IR process, and showed that it can enhance the retrieval process. Retrieval benefits were small for short (more realistic) queries. In this respect (looking at performance metrics only), our work confirms the general consensus that linguistics is of limited use to realistic IR systems. However, our proposed term weight is significantly beneficial for longer queries, indicating that it could be used on a document basis, for instance summed over all document terms to indicate how informative documents are in general. This information may be of use to IR, e.g. when matching documents to queries, to boost the relevance score of generally informative documents, or when indexing documents before matching, to indicate poor-content documents not worth indexing (with storage and processing speed benefits). In addition, our term weight could be used in other automatic language processing applications, e.g. summarisation, to indicate informative words or passages. Overall, the take-home message is that linguistic applications to IR that bring small benefits should not limit the scope of research.

Christina Lioma <christina.lioma@cs.kuleuven.be>
C.J. Keith van Rijsbergen <keith@dcs.gla.ac.uk.>

References

Bas, A., Denison, D., Keizer, E. & Popova, G. (eds.) (2004). *Fuzzy Grammar, a Reader*. Oxford, Oxford University Press.

Brown, P.F., Della Pietra, V.J., deSouza, P.V., Lai, J.C., Mercer, R.L. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18, 467-479.

Blanco, R. & Barreiro, A. (2008). Probabilistic document length priors for information retrieval. In *Proceedings of the 30th European Conference on Information Retrieval (ECIR)*, Glasgow, Scotland (in press).

Damerau, F.J. (1971). *Markov Models and Linguistic Theory*. Janua linguarum series minor 95, The Hague, Mouton.

Feller, W. (1950). *An Introduction to the Probability Theory and its Applications*. New York, Wiley.

Gallager, R.G. (1968). *Information Theory and Reliable Communication*. New York, Wiley.

Good, I.J. (1968). *The Estimation of Probabilities: an Essay of Modern Bayesian Methods*. Cambridge, MIT Press.

Jespersen, O. (1913). *Sprogets Logik (The Logic of Language)*. Copenhagen, University of Copenhagen.

Jespersen, O. (1929). *The Philosophy of Grammar*. London, Allen and Unwin.

Karlgren, J. (1993). Syntax in information retrieval. In *Proceedings of the 1st Nordic Doctoral Symposium on Computational Linguistics,* Copenhagen, Denmark.

Kraaij, W., Westerveld, T. & Hiemstra, D. (2002). The importance of prior probabilities for entry page search. In *Proceedings of the 25th ACM SIGIR Conference on Research and Development in Information Retrieval*, Tempere, Finland, 27-34.

Lioma, C. (2008). *Part of speech n-grams for information retrieval*. PhD thesis, University of Glasgow.

Lyons, J. (1977). *Semantics.* Vol. 2. Cambridge, Cambridge University Press.

Marcus, M.P., Santorini, B. & Marcinkiewicz, M.A. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19, 313-330.

Ounis, I., Lioma, C., Macdonald, C. & Plachouras, V. (2007). Research directions in Terrier: A search engine for advanced retrieval on the Web. *Novatica/UPGRADE The European Journal for the Informatics professional. Special Issue on Web Information Access, 49-56*.

Porter, M.F. (1980). An algorithm for suffix stripping. *Program*, 14, 130-137.

Robertson, S. & Walker, S. (1994). Some simple approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval*, Dublin, Ireland, 232-241.

Schmid, H. (1997). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the conference on New Methods in Language Processing Studies*, Manchester, UK, 154-164.

Smeaton, A.F. (1986). Incorporating syntactic information into a document retrieval strategy: An investigation. In *Proceedings of the 9th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, Pisa, Italy, 103-113.

Smeaton, A.F. (1999). *Using NLP or NLP Resources for Information Retrieval Tasks. Natural Language Information Retrieval.* Dordrecht, Kluwer.

Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application to retrieval. *Journal of Documentation*, 28(1), 11-21.

Tait, J.I. & Oakes, M. (eds.) (2006). Proceedings of the workshop: How can computational linguistics improve information retrieval? In *Proceedings of the CLIIR Workshop at the International Conference on Computational Linguistics-Association for Computational Linguistics (COLING-ACL) 2006*, Sydney, Australia, v (preface).

Tait, J.I. (ed.) (2005). *Charting a New Course: Natural Language Processing and Information Retrieval. Essays in Honour of Karen Spärck Jones*. The Information Retrieval Series, Vol. 16, Berlin, Springer Verlag.

APPENDIX I

Derivation of the probability that a second degree part of speech is informative, using Bayes rule for combining probabilities (see Section 4.1).

Let $\text{inf} = X$, $pos' = Y$, and $pos'' = \overline{Y}$.

Then, Equation 5, $P(\text{inf} \mid pos') = \lambda$, can be written as $P(X \mid Y) = \lambda$

and Equation 6, $P(\text{inf} \mid pos'') = \rho$, can be written as $P(X \mid \overline{Y}) = \rho$.

Using Bayes theory, the following holds:

$$\frac{P(X \wedge Y)}{P(X)} = P(Y \mid X) \text{ and } \frac{P(X \wedge Y)}{P(Y)} = P(X \mid Y)$$

It follows that: $P(X \mid Y) = \lambda = \frac{P(Y \mid X)}{P(Y)} P(X) \Rightarrow P(Y \mid X)P(X) = \lambda P(Y)$

Then, ρ is derived as follows:

$$\rho = P(X \mid \overline{Y}) = \frac{P(\overline{Y} \mid X)}{P(\overline{Y})} P(X) = \frac{[1 - P(Y \mid X)]}{1 - P(Y)} P(X) = \frac{P(X) - \lambda P(Y)}{1 - P(Y)}$$

By setting $\lambda = 1$, we can solve for ρ: $\rho = \frac{P(\text{inf}) - P(pos')}{P(pos'')}$

There can be one pos' event (nouns) and three pos'' events (adjectives, verbs, participles), giving a total of four events. Hence, $P(pos') = 0.25$, and $P(pos'') = 0.75$. Making no assumption about the probability of informative content occurring ($P(\text{inf}) = 0.5$), $\rho = 0.50 - 0.25 / 0.75 \approx 0.33$.