



Swedish institute of Computer Science

The Basics of Information Retrieval

Jussi Karlgren

Contents

I	Information Retrieval	1
1	Information Retrieval: Statistics and Linguistics	3
1.1	Manual vs Automatic methods	3
1.2	Words as indicators of document topic	6
1.2.1	Analyze the document	6
1.2.2	Knowledge about language	7
1.2.3	Combining tf and idf	11
1.2.4	Beyond single word frequency counts	11
1.2.5	Reducing the number of terms: Conflation	11
1.2.6	Increasing the number of terms: Complex terms	12
1.2.7	How can we improve indexing?	14
1.3	Document Variation	14
1.3.1	Document length effects	14
1.3.2	Document Style	15
1.4	Structured text and structuring text	16
1.4.1	Text is more than a bag of words	16
1.4.2	Word distributions are bursty	16
1.4.3	Syntax, Semantics, and Text Studies	16
1.4.4	Fielded search	18
1.4.5	Text Segmentation	18
1.4.6	Passage Retrieval and Question Answering	18
1.4.7	Abstracting and Summarization	19
1.4.8	How textuality could be utilized better	19
1.5	Requests and queries: Dialog	20
1.5.1	Boolean and probabilistic approaches	20
1.6	Information Access Processes: Texts more than are	21
1.6.1	Typical query processing	21
1.6.2	Beyond single queries	22

1.6.3	Query expansion	22
1.6.4	Relevance Feedback	23
1.6.5	Other qualities of text	23
1.7	Texts are sometimes written in other languages than English	23
1.8	The contribution to linguistics	24
1.9	Conclusions: open research questions	24
2	Evaluating information retrieval	27
2.1	How exhaustive is the search? - Recall	27
2.2	How much garbage? - Precision	27
2.3	Combining precision and recall	27
2.4	What is wrong with the evaluation measures?	27
3	References	31

Part I

Information Retrieval

Chapter 1

Information Retrieval: Statistics and Linguistics

Organizing a document collection so that documents can be found easily is difficult, especially if more than one reader is expected to be able to use the collection. This text gives a brief overview of existing automatic methods for text indexing and retrieval – one widely used technology for organizing collections automatically or semi-automatically – and identifies some directions for future research.

1.1 Manual vs Automatic methods

The traditional way of organizing documents and books is sorting them physically in shelves after categories that have been predetermined. This generally works well, but finding the right balance between category generality and category specificity is difficult; the library client has to learn the categorization scheme; quite often it is difficult to determine what category a document belongs to; and quite often a document may rightly belong to several categories.

Some of these drawbacks can be remedied by installing an *index* (from Lat. *index*: forefinger, pointer) to the document collection. Documents can be given several pointers using several methods and can thus be reached by any of several routes. *Indexing* is the practice of establishing correspondences between a set, possibly large and typically finite, of *index terms* or search terms and individual documents or sections thereof. Index terms are meant to indicate the topic or the content of the text: the set of terms is chosen to reflect the topical structure of the collection, such as it can be determined. Indexing is typically done by indexers – persons who read documents and assign index terms to them. Manual indexing is often both difficult and dull; it poses great demands on consistency from indexing session to indexing session and between different indexers. It is the sort of job which is a prime candidate for automatization.

Automating human performance is never trivial, even when the task at hand may seem repetitive and non-creative at first glance. Manual indexing by human indexers is a quite complex task, and difficult to emulate by computers. Manual indexers and abstractors are not consistent, much to the astonishment of documentation researchers. The fact

is that establishing a general purpose representation of a text's content is probably an impossible task: anticipating future uses of a document is difficult.

Typically manual indexing schemes control the indexing process by careful instructions and an established set of allowed index terms. This naturally reduces variation, but also limits the flexibility of the resulting searches: the trade-off between predictability and flexibility becomes a key issue. The idea of limiting semantic variation to a discrete set of well defined terms – an idea which crops up regularly in fields such as artificial intelligence or machine translation – is of course a dramatic simplification of human linguistic behavior. “We note that in natural languages – and their design is successful in this respect – communication normally proceeds without explicit definition of terms. Not only do different persons attach slightly different meanings to the same terms but no person has ever even for himself delimited an exact or definable meaning of terms, except possibly for some few of them ... In normal human communication, introduction of an explicit definition for natural language terms is a symptom of malfunction. One may ask oneself whether natural language succeeds not in spite of but thanks to the absence of rigid definition of meaning. The flexibility of natural language semantics appears also from the observation that definitions of terms age much faster than the terms themselves.” (H. Karlgren, 1976).

By and large computerized indexing schemes have distanced themselves from their early goal of emulating human indexing performance to concentrating on what computers do well, namely working over large bodies of data. Where initially the main body of work in information retrieval research has been to develop methods to handle the relative poverty of data in reference databases, and title-only or abstract-only document bases, the focus has shifted to developing methods to cope with the abundance of data and dynamic nature of document databases today.

This is where the most noticeable methodological shift during the past forty years can be found. Systems today typically do not take the set of index terms to be predefined, but use the material they find in the texts themselves as the starting point: a shift from what sometimes is called *pre-coordinate* to *post-coordinate* indexing. This shift is accompanied by the shift from a set-theoretical view of document bases to a probabilistic view of retrieval: modern retrieval systems typically do not use boolean constraints but rank documents for likelihood of retrieval relevance by term weights rather than term presence. The indexes typically generated by the algorithms today are more geared towards fully automatic systems rather than building index term sets for human perusal. This allows the index term sets to grow larger.

However, the field has not experienced major methodological and theoretical breakthroughs. Basically, the information retrieval systems of today work in an intuitively appealingly simple way, using algorithms about forty years old. Most systems that are deployed for public use today are based on ideas that were known, established, and first explored empirically in the 1950's (Luhn, 1957, 1958, 1959). This is not to mean that the actual retrieval services have not improved strikingly over the past forty years: early conjecture has been solidified into algorithms; algorithms based on early conjecture have been verified mathematically, tested on large corpora, and developed and enhanced since. Systems today can – largely thanks to better hardware – make better use of users to improve their performance. There are full texts available, the interfaces to the systems are faster and better designed, the processing speed is high enough to permit interactive

search – interactive in the sense that the user can be expected to provide the continuity of the dialog process – and the computer literacy of the average reader has increased to the point where enough library clients can be expected to use a computer search system to search and find documents for such systems to be designed and deployed in most libraries in well-to-do neighborhoods.

Text as an object of study

Information retrieval technology is largely about text and the content of text. (Which of course is a limitation which may seem inappropriate in view of the large variety of information sources available to us.) In a research area which mainly concerns text one might expect to find fertile ground to apply results from linguistics, and to find it provides research to the study of language – especially written language. Now, the standard model for information retrieval is roughly as shown in figure 1.1. There is a body of text; information requests are put to a system which handles this body of text; the texts are analysed by some form of analysis procedure to yield a non-textual representation of the same; the information requests are likewise analysed by an identical or similar procedure to yield a query. The two representations are then matched. The texts with the best matches are presented as potential information sources to fulfil the request.

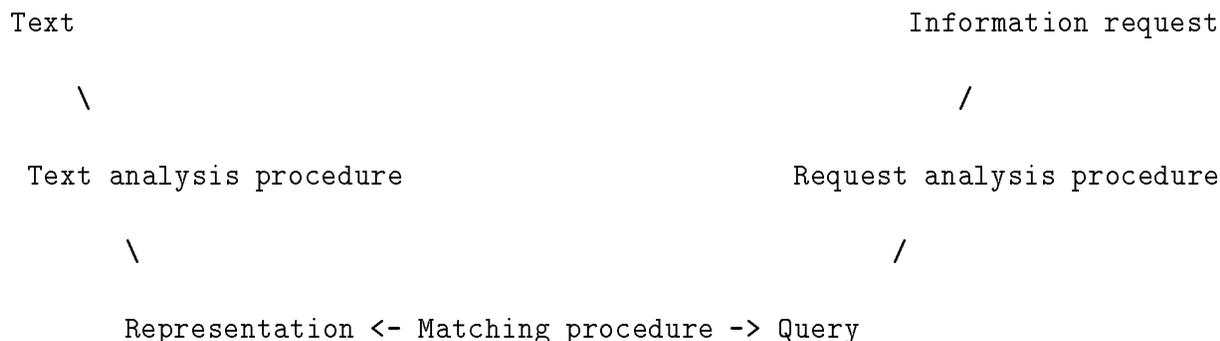


Figure 1.1: The Standard Model of Information Retrieval

In fact very little of this process is actually based on explicit knowledge of language. Typically both analysis procedures and matching procedure are performed using statistical methods. The role for linguistics or knowledge of language in general is usually assumed to be in improving the analysis of requests and texts; the representations are in some way assumed to be a-linguistic and amenable to pure formal manipulation. The point of the analysis operations is typically taken to be a) to reduce the amount of information in order to make the representations manageable – and the noise caused by language and the freedom human languages afford their users are crucially important to reduce to that end – and b) straighten out the vagueness and indeterminacy inherent in natural language in order to facilitate matching.

This quite intuitive and in many ways appealing model hides the complexity of human language use from the matching procedure, which can then be addressed used formal

methods. This is not entirely to the benefit of the enterprise. The very same mechanisms which make the matching complicated – the vagueness and indeterminacy of human language – are what makes human language work well as a communicative tool; awareness of this is typically abstracted out of the search process. The major difference between automated information retrieval and consulting with a human information analyst is that the latter normally does not require the request to be transformed to some invariant and unambiguous representation; neither does the human analyst require the documents to be analyzed into such an representation. A human analyst not only copes with but utilizes the flexibility of information in human language: it is not an obstacle but a feature. A seemingly unrelated text may contain valuable relevant parallels to a request. “Vagueness may be the price that has to be paid in order to achieve the kind of gliding from one concept to another which is necessary for non-trivial retrieval” (H. Karlgren, 1976).

Homeosemy

It is clear we need a better understanding of how semantics are negotiated in human language usage. Fixed representations do not seem practical, and do not reflect human language usage. We need more exact study of inexact expression, of the *homeosemy*, (homeo- from Greek *homoios* similar) or near and close synonymy of expressions of human language (H. Karlgren, 1976). This will become all the more clearer if we raise our perspective beyond that of text retrieval, to attempting retrieval of non-textual documents.

1.2 Words as indicators of document topic

The basic assumption of automatic indexing mechanisms is that the presence or absence of a word – or more generally, a term, which can be any word or combination of words – in a document is indicative of topic.

The central task in indexing is the choice of index term vocabulary. We will in the following assume that the indexing vocabulary for the most part will be based on knowledge about the vocabulary of the texts, rather than a predetermined set. To understand the vocabulary of the texts, we will need to understand how the language the texts belong to work. Then the task reduces to: how can we pick relevant terms to describe a text, given that we know what terms are in it and how those terms are used elsewhere?

1.2.1 Analyze the document

The first steps to find index terms automatically is to build a list of words in a text, and calculate their frequency of occurrence. The more frequent terms are considered more valuable in proportion to their observed frequencies. This design suggestion is first made by Hans Peter Luhn (1957, 1959), and the measure is commonly called *term frequency* or, imaginatively, *tf* for short. For this text, for instance, the list will be as shown in table 1.2. Typically, for each document the term weights are collected in a vector – a

term vector – where each position in the vector represents a term, and each position holds the term weight for that document.

92	the
72	of
64	and
62	to
60	a
55	in
51	is
30	for
26	terms
25	documents
24	be
23	that
23	as
22	words
22	term
21	text
20	information
19	document
17	this
17	retrieval
16	are
...	

Figure 1.2: Frequency table of words in this text.

As a semantic representation a term vector formed from a list such as the one in Table 1.2 is poor. An obvious improvement is to filter out certain words that seem to have little to do with topic. A list of such words, most often grammatical form words and other closed class words, is commonly called a *stop list*. Another is to note – as Luhn does in his 1959 paper – that the most frequent words seldom are significant for this sort of enterprise, and that thus it might be possible to filter them out automatically, based on their frequency rather than their text-external or linguistic features.

1.2.2 Knowledge about language

If we try to determine what terms in a document are significant for representing its content, we find that terms that are common in a document, but also common in all other documents are less useful than others. The question is how *specific* a term is to a document, or how uncommonly common it is.

Collection frequency, *inverse document frequency* or *idf* is a measure of term specificity originally defined by Karen Sparck-Jones (1972). *Idf* is a function of N/d_i , where N is the total number of documents in the collection and d_i is the number of documents where term i occurs – the *document frequency*. This measure gives high value to terms which

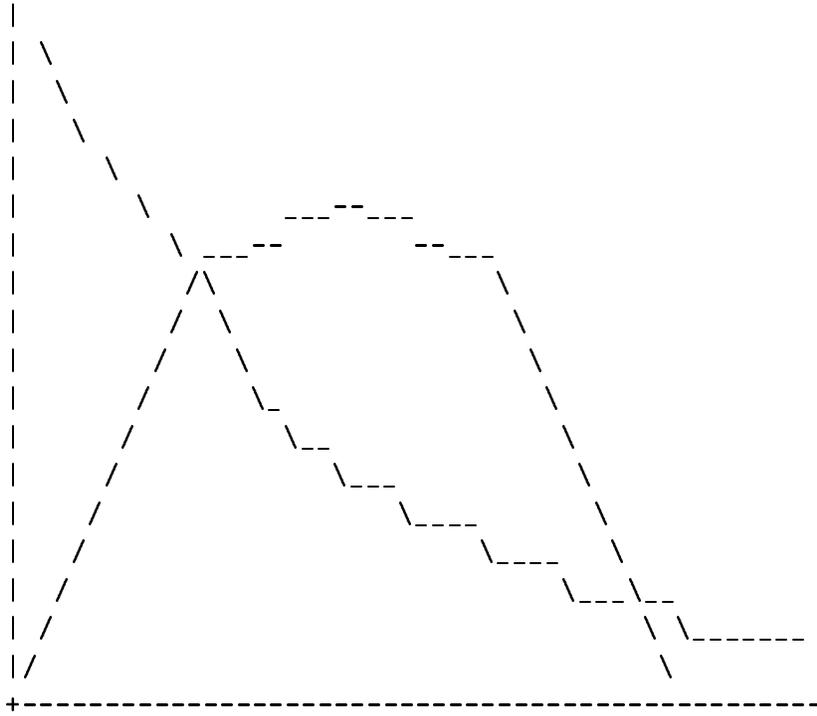


Figure 1.3: Significance vs frequency. From Luhn (1959).

the
a
and
that
one
it
two
may
could
such
next
just
half
both
of
to
in
for
...

Figure 1.4: Stoplist.

```

26 terms
25 documents
22 words
22 term
21 text
20 information
19 document
17 retrieval
14 idf
14 frequency
11 technical
10 word
10 indexing
10 collection
 9 table
 8 single
 8 query
  ...

```

Figure 1.5: Frequency table of words in this text, filtered with stoplist.

occur in only a few documents. Used alone, it gives about as useful results as term frequency used alone – *idf* is vectored towards high precision while *tf* gives better recall or indexing exhaustivity.

There are several modifications of the *idf* measure. One can use paragraphs instead of documents, to model the fact that documents may not be homogenous (Lahtinen, 1998); one may weight the measure in different ways based on the document properties, e.g. as suggested by Tokunaga and Iwayama (1994). Their measure – the *weighted idf* or *widf* – is weighted for term frequency in the documents which it occurs in: the *widf* is calculated as a function of *d_{fi}* rather than *d_i*, where *d_{fi}* is the frequencies of term *i* in the respective documents. Their experiments seem to indicate an improvement in performance – but they have sacrificed some of the probabilistic theoretical underpinnings of Sparck Jones work.

A problem with *idf* as a measure is that it is unclear what universe the document frequency should be calculated over. The calculation depends on an *N*, a total number of documents, and establishing what general usage of a term is may be difficult, if not impossible. In some cases a collection is so well defined that a collection internal *idf* is quite adequate; in others, where potential readers may not be aware of the collection setup or if the collection is very heterogenous it may not. In table 1.6 you will find *idf* scores for words in this document; the scores are calculated with respect to the top twenty documents retrieved by Altavista for the search terms “information, retrieval, algorithm”.

0.019 the	...
0.020 and	1 behavior
0.020 retrieval	1 behaviour
0.020 to	...
0.021 information	1 karlgren
0.021 for	...
0.022 is	1 mathematical
0.022 of	1 mathematically
0.023 a	1 mathematics
0.023 this	...
0.023 with	1 microwave
0.024 in	...
0.025 as	1 miscellaneous
0.026 from	...
0.026 or	1 morphology
0.026 not	...
0.027 search	1 nationwide
0.029 be	...
0.030 use	1 navigation
0.031 but	...
0.031 language	1 pulmonary
0.032 example	...
0.033 form	1 radio
0.033 on	...
0.033 text	1 redundancy
0.033 web	...
0.034 documents	
0.034 first	
0.034 queries	
0.034 query	
0.034 words	
...	

Figure 1.6: Inverted document frequencies for terms in this collection.

1.2.3 Combining *tf* and *idf*

There are various ways of combining term frequencies and inverse document frequencies, and from empirical studies (e.g. Salton and Yang, 1973), we find that the optimal combination may vary from collection to collection. Generally, *tf* is multiplied by *idf* to obtain a combined term weight. Alternatives would be for instance to entirely discard terms with *idf* below a set threshold – which seems to be slightly better for high precision searches.

1.2.4 Beyond single word frequency counts

So far, the methods outlined above use knowledge of language only indirectly. But linguistic methods have obvious roles to play for index term selection. One reason to apply linguistic knowledge to index term selection is to provide multi-element terms effectively. This is assumed to provide gains in precision, by allowing finer grained distinctions between similar but non-identical multi-element terms with differing internal structure, or by establishing more elaborate relations between identified term elements. Thus, it would be possible to distinguish between representation-wise similar but non-identical documents.

Another reason is to conflate similar variants into one index term. This is assumed to provide gains in recall, by allowing more documents with only trivial differences to be keyed by the same set of terms. The first has typically involved research in syntax, word dependencies, derivational morphology, and terminology; the second in inflectional morphology. So far, Sparck Jones finds no conclusive improvement from using either technique has been established (1997). All of these techniques can more or less be approximated using purely statistical methods.

1.2.5 Reducing the number of terms: Conflation

Morphological Conflation

As can be seen in tables 1.2 and 1.5 the words “document” and “documents” both show up in the beginning of the list. The words “indexed” and “indexing” do not, and probably should – they show up further down in the list. Word form analysis, or *morphological analysis* would conflate these forms, and raise their combined weight.

Morphological analysis to identify morphological variants of a lexeme are normally implemented as *stemming* or simple suffix stripping. Porter (1980) describes a widely adopted and efficient context-sensitive stemming algorithm for English based on a suffix list. Alternatively the user can be encouraged not to enter full words but truncated forms.

The utility of stemming for English is debatable, (Harman, 1991) but its intuitive merits are good enough and its cost in processing quite low, so many systems make some effort in this direction. “This means that matches are not missed through trivial word variation as with singular/plural forms.” (Robertson and Sparck Jones, 1996). English, of course, has an exceedingly spare morphology, with few morphological variants and tends

to not form graphical compounds as often as other languages: both these characteristics would seem to decrease the utility of an elaborate morphological analysis.

It is unfortunate for the generality of the results in the field that the research and business language of the world currently is English. Simple stemming is sufficient for English, but not for most other languages of the world. In comparison, where experiments on morphological analysis based normalization on material from languages other than English have been performed, they do provide improved results: how, and exactly what is useful depends on the language. (Slovene: Popović and Willett, 1992; Finnish: Koskenniemi, 1996; Dutch: Kraaij and Pohlmann, 1996; French: Jacquemin and Tzoukermann, 1998).

Synonyms or semantic conflation

Another aspect of conflation is finding sets of synonyms – such as they may exist – or near synonyms, and equating them for search purposes. This is typically done with a static word list – a *thesaurus* (from Greek xxx, rich, as it were) – based on compiled lexical knowledge.

Alternatively there are statistical techniques which reduce a large set of words to a smaller set of senses, most notably Latent Semantic Indexing (Deerwester et al, 1990). Latent Semantic Indexing works from the observation that a matrix of index terms by documents is sparse: most terms do not appear in most documents. This matrix can be reduced to a smaller, and thus denser, matrix by various mathematical technique, e.g. singular value decomposition, which will conflate terms with very similar distributions. The resulting entries are in some sense senses. How much one wants to reduce the matrix is a question of how much information one is willing to lose to gain the better recall given by the conflation.

1.2.6 Increasing the number of terms: Complex terms

Multi-word terms

Counting solitary words is fine, but the idea that lone words by themselves carry the topic of the text is one of the more obvious over-simplifications in the model so far. Indexing texts on ice cream on “ice” and “cream” is intuitively less useful than looking at the combination “ice cream”. However, in experiments designed to test the usefulness of multi-word terms, any addition past single word indexing is cumbersome and expensive in memory and processing, while adding comparatively little to performance. In any case, the discriminatory power of single word terms is much stronger than that of any other information source (Strzalkowski et al, 1997).

Finding multi word terms can be done by statistical techniques or by linguistically motivated techniques.

Collocations and Multi-word technical terms

One way of expanding the search to words beyond single terms is simply tabulating words that occur adjacently in the text – *n-grams*. For instance, Magnus Merkel has implemented a tool for retrieving recurrent word sequences in text (1994).

Using more theoretical apparatus, other types of arbitrary and recurrent combinations in the text – *collocations* – can be recognized and tabulated as well. Frank Smadja has implemented a set of tools (1992) for retrieving collocations of various types using both statistical and lexical information; he identifies three major types of collocations: predicative relations such as hold between verbs and their objects in recurrent constructions, set noun phrases, and phrasal templates, where only a certain slot varies from instance to instance.

To extract collocations of the second type, Justeson and Katz (1995) have added lexical knowledge simple statistics, and use it to extract *technical terms*. Technical terms are a specific category of words which behave almost like proper names. They cannot easily be modified – their elements cannot be scrambled or replaced by more or less synonymous components, and they usually cannot be referred to with pronouns. Thus, the technical terms tend to stay invariant throughout a text, and between texts.

Justeson’s and Katz’ appealingly simple algorithm to spot multi-word technical terms tabulates all multi-word sequences with a noun head from a text, and retains those that appear more than once. This method gives a surprisingly characteristic picture of a text topic, given that the text is of a technical or at least non-fiction nature. Their major point is well worth noting: the fact that a complex noun phrase is used more than once identically is evidence enough for its special quality as a technical term. It is repetition, not frequency, that is notable for longer terms.

Linguistic methods are often suggested for the purpose of extracting multi-word terms. Indeed, “The main modern rationale for linguistically motivated indexing is in capturing multi-element terms effectively.” (Sparck Jones, 1997). The research in linguistically motivated indexing has typically taken statistically generated multi-word terms as a baseline and attempted to identify better terms. An example is Strzalkowski’s work in trying to find linguistically motivated content word combinations through statistical analysis of word pairs and the dependence relation between them (1994). Strzalkowski has experimented using head modifier structures from fully parsed texts to extract index terms: this normalizes phrases such as “information retrieval” and “retrieval of information” to the same index representation.

However, it has been repeatedly shown that compound terms do not improve retrieval performance more than marginally, (Fagan, 1989) and that the effort needed to implement and run linguistic methods in general is not worth the gain (Sparck Jones, 1997). On this note, Robertson and Sparck Jones discourage implementers from considering other than previously known multi-element terms: “Discovering, by inspection, what multi-word strings there are in a file is ... a very expensive enterprise. ... In general these elaborations are tricky to manage and not recommended for beginners.” (1996).

1.2.7 How can we improve indexing?

Indexing seems to work quite well as it is – but the results of systems based on the mechanisms outlined in this section can be improved. Not because of any obvious drawbacks in the mechanisms themselves: they provide consistent and stable results, with variation from system to system surprisingly small; the reason to continue work is that the stable results are not only consistent but consistently mediocre. Word based indexing is not sophisticated enough as a semantical model to capture important facets of textual variation. We need to investigate how to create and make use of several different indexing methods simultaneously. So far, most work along this vein – *merging information streams* – has been done manually (Strzalkowski et al, 1996); more efficient would be develop automatic merging schemes. Ideally one would want streams which perform differently from each other: in practice so far, most streams seem to make the same sort of misses.

However, a stronger case for continuing experiments on indexing schemes even in the face of the reasonably stable results obtained to date, is the fact that no substantive research has been performed on other than English text. English is a typologically special language in that it relies more on word order than on inflection than most other languages; this can be expected both to decrease the value of normalization through morphological analysis and the utility of linear precedence based statistical metrics. If we can expect words to appear adjacently in a predictable order with minimal variation from occurrence to occurrence the systems we build will be very different than if we assume there are long range dependencies between haphazardly appearing words marked with agreement features.

While indexing technology typically is vectored towards automatic retrieval, building indexes for human use can still be useful. As a chart over a document database, using a set of index terms users may get a picture of what the database contains (Hulth and Jonsson, 1999). But the requirements for such an indexing scheme are vastly different than those for automatic processing, and the difference has hardly been studied.

1.3 Document Variation

1.3.1 Document length effects

As the term weight is defined in the *tf* component of the combined formula, it is heavily influenced by document length. A long document about a topic is likely to have more hits than a short one will for a relevant term; this may not reflect its likelihood of being relevant.

Most algorithms in use introduce document length as a normalization factor of some sort, if the documents in the document base vary as regards length (Salton and Buckley, 1988). The most frequently used one is the *cosine formula* which originates from the SMART system. It reduces each term weight in the document vector of a document *d* by dividing it with $\sqrt{(\sum_{i=1}^N tf(\text{document}_d, w_i))}$. This gives quite a strict normalization: it promotes short documents disproportionately, and in practice the effects of the cosine formula have to be damped somewhat, as long documents often turn out to be more

interesting than what it would assume (for a discussion of various formulations of the cosine formula, see e.g. Singhal et al, 1995).

Another formula to reduce the effect of document length is the OKAPI formula. It divides each term weight with a factor derived from the document length in words; the strength of the reduction is controlled using a parameter which is set after experimenting with the collection at hand (Robertson and Sparck Jones, 1996).

1.3.2 Document Style

Texts are so much more than just sets of words. Indeed, texts are more than just what they are about. Texts vary in many ways. Authors make choices when they write a text: they decide how to organize the material they have planned to introduce; they make choices between synonyms and syntactic constructions; they choose an intended audience for the text. Authors will make these choices in various ways and for various reasons: based on personal preferences, on their view of the reader, and on what they know and like about other similar texts.

A *style* is a consistent and distinguishable tendency to make some of these linguistic choices. Style is, on a surface level, very obviously detectable as the choice between items in a vocabulary, between types of syntactical constructions, between the various ways a text can be woven from the material it is made of. It is the information carried in a text when compared to other texts, or in a sense compared to language as a whole. This information – if seen or detected by the reader – will impart to the reader a predisposition to understand the meaning of text in certain ways. Or, more roughly put, style is the difference between two ways of saying the same thing.

So, the variation in a text or differences between texts that is not primarily topical, that has not to do with meaning, is stylistic. Naturally, demarcation of stylistic variation to topical variation is impossible. Certain meanings must or tend always to be expressed in certain styles: legal matters tend to be written in legal jargon rather than hexameter; car ownership statistics in journalistic or factual style. The impossibility of drawing a clean line between meaning and style has led to much browbeating among stylisticians and linguists, and discussion about if there in fact are styles at all (see e.g. Enkvist, 1973).

For the purposes of information retrieval, it is in fact all the more interesting to investigate the workings of stylistic variation if it is not completely divorced from topical variation. The purpose would be to find methods to *complement* topical information retrieval, not by improving topical recall nor necessarily topical precision, but by improving the likely subjective quality of the retrieved documents.

The variation that will be most useful to complement topic-based information retrieval is not variation between authors, nor between individual texts, but the consistent, predictable, and distinguishable variation that sets of text may show. The goal is to look for textual characteristics that are measurable quantities – *stylistic items* – and use them to posit variables to categorize or sort texts. The aim is to find *functional styles* (cf. Vachek, 1975) that can be used to understand which *genre* a new or hitherto unknown

text belongs to, and thus to predict the likelihood of the text being interesting to the reader, given that its topic has been determined correctly by topical analysis.

1.4 Structured text and structuring text

1.4.1 Text is more than a bag of words

Text is more than the set of words in it, and specifically, it is more than a plain sequence of words. While texts at first glance are one-dimensional entities, in that linguistic objects follow each other in a sequence, relations between referents, terms, words, entities, subtexts, segments, clauses, paragraphs – or whatever other type of thing one wishes to postulate as suitable items of study – are present in the text and can range quite far over the length of the text. This gives texts a fractal nature of sorts, a character of reaching beyond the one dimension the string affords it. Discovering these relations is largely what text understanding is about. A series of different techniques try to organize the text material into chunks larger than terms. The relations between textual items can take a number of forms, much dependent on the form of analysis chosen.

For text retrieval the standard model discussed so far has appealingly simple and intuitively understandable properties. Find out what a document is about, and find out what the user wants, and match the two; a document is about what is mentioned in it; what is mentioned is mentioned using words; count and tabulate the words. This is easy to understand and to implement.¹ But this simple model has its faults.

1.4.2 Word distributions are bursty

For instance, most statistical approaches assume words appear more or less randomly in a text, in a Poisson-style distribution. This is naturally a gross simplification: words appear in a text not in a memory-less distribution but following a pattern governed by the textual topic progression and communicative conventions (H. Karlgren, 1976; Katz, 1996). If text segments more likely to be topically pertinent are chosen and terms within them weighted up as compared to terms from other sections this weighting would reflect the topical make-up of the text better than a non-progressional model. These sections cover some existing techniques potentially useful for this, such as summarization, text segmentation.

1.4.3 Syntax, Semantics, and Text Studies

Now, linguistics has for the past century busied itself with finding speaker- situation- and meaning-independent models for syntactic analysis and description and speaker- situation- and local-context- independent models for semantic analysis and description.

¹From personal experience I know that a class of computer science students can be taught to understand, appreciate and implement a working information retrieval system from scratch in less than one day.

The former goal can be claimed to have been reached, at least to the extent that existing syntactic models seem close to covering a large proportion of well-edited literary text. The latter seems elusive: there is no established semantic theory with any kind of substantial coverage or generality.

And research on larger units of language use such as texts, dialogs or discourse in general has not succeeded in providing generalizable results. The goal is less concrete: texts are not regular in the sense sentences are, and when formalization is attempted, it only succeeds in prototypical cases. Still, there is reason for optimism. With large amounts of texts available for automatic analysis, linguists can test, discard, verify, and refine methods for large-scale analysis with the same efficiency clause-level analysis was performed earlier. “We take heart particularly from two facts: first, linguists are turning their attention more and more to larger units of discourse than the sentence, and second, on-line retrieval systems are likely to involve retrievable units smaller than traditional documents. We believe that the relevance of these fields to one another will become more apparent as the size of the text units they deal with becomes more commensurable.” (Sparck Jones and Kay, 1976).

However, studying topical progression in a text is complex. Local effects – the distinction between given and new information in a clause, say – have been studied and partially formally described, but not well enough to be useful for predictive work, which is what information retrieval needs. “It is not easy to identify the topic and focus of a printed sentence, especially in such a language as English, where the surface word order is grammatically bound to a great extent.” (Sgall, 1980). And later experiments cover – by author admission – prototypical cases only. (Hajičová, Skoumalová, and Sgall, 1995). There is a systematic problem in automatic text analysis in that text in itself is a semantic object, and has transcended much of the syntactically governed constraints that clause structure adheres to. Text should be analyzed as such. Surface cues are only incidental traces of semantic linking of text (Källgren, 1978, 1979).

Further, texts have many kinds of properties besides being word containers. This is at its most obvious if one instead of text files looks at documents with pictures, at music retrieval, at video retrieval, or image retrieval. These types of document can be categorized in numerous ways, none of them independent of each other. Likewise, albeit less obviously, for texts. Indeed, the fact that text consists of readily identifiable words with obviously regular local dependencies to each other could be said to have lead information retrieval up the impractical path of compositional semantics.

And when further modalities come into play, a more general view must be taken. For instance, some experiments with audio database indexing involve not only a textual representation of the spoken data, but type of dialog. (Kimber et al, 1995; Oard, 1996) Whatever dimensions of variation one accepts as valid for an area or a set of texts, it is clear that a mono-modal text representation – whatever it is, and however well it is designed – simply will not be able to capture more than very simple characteristics of a text, and thus will ultimately constrain the utility of the matching functionality.

1.4.4 Fielded search

Some materials are structured to begin with. A search in a database for the yellow pages, for instance, will naturally make use of free text search as well as separate searches for company names or addresses in separate fields. A search in a press archive will naturally allow for searching in the byline and date fields separately from searching in the text itself. Or should.

1.4.5 Text Segmentation

But most materials are not organized beforehand. It is to some degree possible to assume a structure for textual material which is not explicitly organized. Typically, such analysis is done without regard to likely search requests under the assumption that there is a structure which is possible to chart by inspection of texts in isolation. To some degree this is true, although for instance in text segmentation tasks human subjects do not always agree on where segment boundaries can be assigned (Passonneau and Litman, 1993, found that subjects did agree; Hearst, 1994, found they did, more or less, within a range. Passonneau and Litman used spoken material, and Hearst used written popular science texts. Most likely the richer information in the spoken mode accounted for the difference in results.).

Texts can be split up in subtopic segments based on word occurrences: if word frequencies shift noticeably from one stretch of stretch to another, it is reasonable to assume that there is an attendant shift of topic. This assumption underlies several algorithms for text segmentation. (Hearst and Plaunt, 1993; Hearst, 1994; Reynar, 1994; Salton, 1994; Hearst, 1997). This is the underlying assumption of most text segmentation algorithms (see below.)

1.4.6 Passage Retrieval and Question Answering

Sparck Jones and Kay write 1971 that "... there is little doubt that it is from this direction [fact retrieval or question answering] that many of the new ideas introduced into documentation over the next few years will come." This promise seems from today's perspective not to have been fulfilled. The optimism of the early seventies for solving artificial intelligence and knowledge representation issues was clearly unfounded. It is clear that – similarly to text segmentation – passage retrieval is non-trivial even for humans, even when the data set is quite small.

However, there are some more modest treads along the path to systematic fact retrieval that actually have proved both promising and useful. In general, the idea that a system can find information in text by extracting structures that originate from the information requests themselves is much more tractable than attempting to organize texts in anticipation of future requests.

As an example, algorithms for entity spotting, starting with person, place, and organization name spotting and date spotting to more general entity spotting function with some degree of success (Strzalkowski and Wang, 1996), and add noticeably to information retrieval performance when combined with less inventive single and multi word term

information. Similarly, technical terms have a more rigid structure than other multi-word terms – rather similar to names, in their linguistic behavior, in fact – and can be picked out through pattern matching techniques augmented with lexical information from a part-of-speech lexicon (Justeson and Katz, 1995).

This type of technique can be extended quite far to perform information extraction. Matching recurrent patterns for certain predictable pieces of information can be done with a useful level of accuracy and speed, such as is demonstrated in the yearly Message Understanding Conferences. These techniques are typically based on stereotyped information requests and elaborate linguistic variant detection algorithms that are pre-compiled to simple pattern matches (e.g. Grishman, 1996).

Real live semantic analysis of course is beyond the scope of automatic systems today, but systems for higher level textual analysis are already at the point where the inclusion of semantic knowledge such as precomputed general concept hierarchies such as Wordnet, or of well typed domain-specific selectional restrictions (Grishman and Sterling, 1990) improves extraction results.

1.4.7 Abstracting and Summarization

A common problem in information retrieval is that there is a large number of documents which may be relevant and may be not, and that deciding which are which is time-consuming. For this purpose, automatic abstracting, summarization or gisting algorithms attempt to provide a compact version of a text.

Most automatic abstraction algorithms work on the assumption that selecting a number of sentences from the text will provide a picture of the text topic progression (Luhn, 1958).

1.4.8 How textuality could be utilized better

Texts do have structure – that much is evident. So far, little of this structure has been used explicitly for information retrieval. There are numbers of experiments that wait to be performed: if a text can be structured by some means, and its components indexed separately, such a composite index might well provide a richer picture of text topic than a simple list. Clause weighting approaches, topic-focus detection, foreground-background clause identification, summarization, and subtopic segmentation are all techniques available for experimentation: these show promise to perform differently from the single word and multi-word term frequency based indexing schemes detailed in the previous section.

Understanding more of why texts are texts rather than word containers, and why texts in important ways are more like pictures than dictionaries will give more depth to text analysis. The objective is some level of topical or semantic analysis, and from the discussion above and in the introduction, it seems abundantly clear this should be performed in interaction with the intended reader of the text. The reader or user is not a single one-shot question - the user is accessing text for some reason, and this reason is not irrelevant for information retrieval purposes.

1.5 Requests and queries: Dialog

The preceding sections have concentrated on document analysis. Central to the enterprise of searching document databases is the information need, as experienced by the user. These sections will outline how the information need relates to a document database.

1.5.1 Boolean and probabilistic approaches

The basic scenario of traditional *information retrieval* is then finding the correct combination of search terms combined together in a logical structure, often using Boolean operators such as AND OR or NOT, and then examining the set of documents that fit the request. The documents in the set match the request, the documents outside do not. If a document matches a request it is presented, if it does not, it will not be. The most simple set algebraic approach works reasonably well if the set of index terms is limited by design or alternatively for document bases where the documents are short and concise: a list of literature abstracts or document titles, for instance which as a side effect that the set of available terms is limited. It has several good sides: it is easy and efficient to implement and theoretically comprehensible through the well understood properties of set theory.

However, Anselm Spoerri (1994) shows in an example how Boolean search can be difficult to work with. In an example database of several tens of thousands of items on computer science, he poses a query to retrieve items on “visual query languages for retrieving information and that consider human factors issues”. He formulates a Boolean query:

```
(graphical OR visual)
information retrieval
query languages
human factors
```

This query is ambiguous. If AND is used to conjoin the four terms, the query is very restrictive. For this query, Spoerri retrieved one single document in the database. If OR is used, the query is not very specific: for this case, Spoerri obtained 19691 documents. This shows how Boolean search methods can have unexpected results in spite of its theoretically attractive predictability, and that the OR and AND of Boolean logic are deceptive in that they invite comparison with the “or” and “and” of everyday discourse. Most importantly, Boolean methods have no built in way of handling uncertainty.

Boolean systems are widely in use, especially for trained documentalists; for untrained users, the Boolean approach has been largely abandoned in full-text retrieval systems – although the name *retrieval* has been retained for an activity which only in its extreme cases resembles retrieval – in favor of *probabilistic retrieval*, which ranks the retrieved documents by likelihood of providing relevant data for the resolution of the information request as expressed by the search terms. The basic improvement is the *weighting* of terms by assumed importance for a document’s representation. This provides a model of uncertainty, as well as obviates most of the need for a specific query language, at the price of lessening predictability, efficiency, and, to some extent, expressiveness of the interface.

1.6 Information Access Processes: Texts more than are

We have in the above assumed that people will walk up to an information system and state their information need in some consistent manner. This assumption is oversimplified. While it has led to useful generalizations in the design of system innards, the interaction with systems is not always all that could be hoped for.

The typical view of system use is that a user poses an information request, receives a set of documents, selects some of them for further processing, examines some in detail, and finally orders some for delivery. At each interaction point, the user may back up to refine or modify the original request. This type of model is proposed by Oard (ref, doug?) and shown in Table 1.7. Systems today typically follow this model from left to right, in that they always expect interaction to start with some form of specification of an information need, in the form of a small set of topical terms. This will give the user a ranked list of documents, which can be used either to select individual documents for further perusal, or discarded, in order to improve the original query.

The model allows for more types of access, as indeed it should. The prototypical information access scenario covers only parts of typical information needs: several different models and studies show how users behave in various ways depending on what task they are working on. For instance, Belkin analyzes information seeking behavior into four prototypical tasks: searching for a known item, scanning through a list for potentially interesting material, reducing a large number of potentially interesting items to a smaller number, or examining a certain document to verify its qualities (ref, Nick?).

Oard's interaction model - or others like it - readily admits various types of information seeking behavior, and gives pointers to where interaction with a system can be understood as a subsystem of its own; Belkin's task oriented prototypical behaviors give an understanding of what the bottlenecks of information access systems can be.

For instance, it is not necessary to limit oneself to one entry point in the model. One may well envision cases where the starting point is a set of documents, inventively displayed, or segments or bits of one single document. And there is any number of interesting transition between different activities in the model to allow for better interactivity in an information access system. Today's systems mostly are not designed to support other than backtracking in the basic left-to-right model.

Activity	Specification	Visualization	Assessment
Example	Input query terms	Inspect ranked list of documents	Scroll up and down in a document
Development	Beyond words	Beyond ranking	Information refinement

Figure 1.7: A model of interaction points with an information access system.

1.6.1 Typical query processing

In the standard model the information request, posed in English or as a set of search terms, is treated much like the documents in the database: it is analyzed on the basis of

term occurrence, and is transformed into a vector of term weights – for which the term *query* typically is reserved – similar to the term vectors computed for the documents.

Given a query and a document term vector and a query, however the term weights are computed, the question is how to match the two term vectors to find documents that fit the request. The simplest approach is to use the conventional scalar product of the two vectors, and simply add the pairwise products of each the weight of each term under consideration, and sort the documents according to their respective score. Most systems use variations of this method.

Documents and information requests are typically very different. Luhn’s original model was for the searcher to compose an essay of approximately the same form as the sought for documents. In practice, as has been established both by informal observation and several formal experiments, information requests to information retrieval systems tend to be very short; the majority being of three words or less (Rose and Stevens, 1996; Rose and Cutting, forthcoming). This gives very little purchase to most linguistically oriented methods, and one would wish to find methods which would encourage searchers to produce longer requests using more terms (for a very simple, yet successful attempt, see Karlgren and Franzen, 1997).

Given that requests are of different length and different type than the target documents, the way the respective term vectors are to be treated should be different. Most systems use a binary frequency calculation for query terms: occurrence, rather than frequency is used as a basis for weighting (Salton and Buckley, 1988).

1.6.2 Beyond single queries

The rise in response speed and interactivity has given users the possibility of searching by a sequence of queries. The first systems did not make use of sequences – they take a sequence of queries not as a dialog but as a sequence of one-shot requests. “Like so many other kinds of self-’service’, from supermarkets and filling stations upwards, [recent full text search systems] have been promoted by salesmen who style the absence of service as quickness and the user’s labour as automation” (H. Karlgren and Walker, 1980).

Similarly to any type of dialog system, information retrieval systems should provide for persistent and modifiable dialog objects: the previous turns in the discourse should not just go away from query to query. The recent formulation of a query, and the documents retrieved for it should be available for backtracking and for reference during the dialog.

1.6.3 Query expansion

One method to get more textual material for fleshing out a short request is to submit the short request to the system, use the first few retrieved documents as a renewed information request, construct a new query from them, and hope that the first few documents indeed are relevant. If the initial search is focused on precision this is a way of improving recall (Strzalkowski et al, 1997).

1.6.4 Relevance Feedback

Alternatively, a retrieval system can present the list of retrieved documents to the user, and have users note which documents seem useful at first glance. These relevant documents are then used to generate a new query. Analogously, non-relevant documents can be discarded in the first iteration, and the terms in them weighted down in subsequent iterations. This technique – *relevance feedback* – was first formulated by Salton.

Some researchers have doubted the usefulness of the technique – especially negative feedback, the exclusion of terms culled from discarded documents to increase precision, can lead to surprising results for the hapless user. However, in user tests it seems to work quite reasonably (Koenemann and Belkin, 1996), and has been formalized usefully for simple implementation (Robertson and Sparck Jones, 1996).

Relevance feedback can be extended by clustering the retrieved documents in similarity sets, if the system has an efficient clustering algorithm. Cutting et al have implemented the *Scatter/Gather* algorithm which does this (1992). Users can select not only single documents for relevance feedback, but entire clusters of documents, represented by terms common to the entire cluster.

1.6.5 Other qualities of text

Texts are used, and often used systematically. When any certain text is read, certain other texts are likely to be read as a consequence, or to appear in its vicinity in some way. This is something that can be used to study text ecology, or the social characteristics of text (Walker, 1981; Belkin, 1994). In the last few years, interest in utilizing text usage as another indicator of text usefulness has resulted in a number of studies and indeed a number of implementations which recommend items to users based on their previous access habits. (Karlgrén, 1990, 1994; Resnick, 1994; Shardanand and Maes, 1994; Hill, 1994).

1.7 Texts are sometimes written in other languages than English

As has been argued above, typological bias renders most discussions of the utility of linguistically motivated indexing moot. Non-linguistic and linguistic methods alike have been tested on English texts. English is a typologically special language. It relies more on word order than do most languages, and its morphology is more impoverished than most. These characteristics have effects not only on the linguistic methods but on the design of purely statistical algorithms as well. If linear order is important, collocations can be assumed to simpler than if long distance relations are marked by agreement markers of some sort.

And more urgently, texts written in a language the reader does not comprehend risk being ignored for the wrong reasons. While fully automatic general purpose high quality machine translation remains a perpetuum mobile, special purpose translation machinery

does show promise of usefulness. Especially if the distinctions between crude, raw, and skim-only-translations (H. Karlgren, 1987) are made clear to system providers, we may expect to be able to peruse texts usefully in languages we do not master.

But we will still need to retrieve them. Cross-linguistic retrieval systems can be built using several different methods. The query itself may be translated. The document representations may be translated. Or the document representations and queries may both be represented in a common language: recent experiments use Latent Semantic Indexing for that purpose.

So multi-lingual retrieval needs to be explored, for the obvious reason that interesting material may be available in the wrong language. Equally crucially, multi-lingual retrieval may improve retrieval in general by clearing the decks from the linguistic bias of results so far.

1.8 The contribution to linguistics

Almost no results from information retrieval research seem to interest linguists. Given that linguistics focuses on the theory of clause structure, and information retrieval on appearance of words and texts, this may not be entirely surprising. However, the application of statistics on large bodies of language data itself is a form of study of language. The information found is not in an explicit form, but if a result from practical systems is that two content words within a four word span from each other tend to form content-bearing associations where longer spans do not, this in itself ought to be interesting for the study of language; if latent semantic indexing works in finding generalizable topical clusters of documents irrespective of the language they are written in, this in itself ought to be interesting for the study of language; if retrieval of admittedly shoddy output from speech recognition systems works on average as well as retrieval of carefully proof-read texts, this in itself ought to be interesting for the study of language. But results such as these are not appreciated by linguists or information scientists, for other than motivation for engineering efforts.

Partly, this will have to do with the exceedingly simple view of language taken by information scientists: documents consist of words, and the words can be used for determining what the document is about. This is only true to a limit, as can be inferred from the results shown in previous sections. The simple view uses a misleadingly simple shortcut to analyzing meaning in text. Most likely, text retrieval and text access cannot be understood in any real way until more general questions, e.g. image access have been understood well enough to have been posed.

1.9 Conclusions: open research questions

This text argues for a multi-lingual and multi-modal analysis of texts. Without this, any analysis scheme will not be open-ended enough to stay useful for other than very well trained specialists.

In experiments described in companion texts to this, we have investigated various ways of enriching the representation of texts, of weighing together different types of knowledge, of studying users of texts, and of improving dialog with search mechanisms to cope with such enriched representations.

In future work we propose to add study of other languages than English; more global textual phenomena; more modalities than pure 7-bit text; more aspects of language use such as studies of the practice of human question answering outside laboratories rather than study of models of question answering in model worlds; and studies of how terms are introduced and moved off-stage during the course of discourse.

Chapter 2

Evaluating information retrieval

Information retrieval research has a well defined and well established set of evaluation tools.

2.1 How exhaustive is the search? - Recall

If one has a good estimate of how many relevant documents a document base contains for some query, one can calculate the proportion of the total set of relevant documents found and retrieved by an algorithm. This proportion is called *recall*.

2.2 How much garbage? - Precision

The proportion of relevant documents in a retrieved set is called *precision* and is a complement measure to recall.

2.3 Combining precision and recall

Trivially, if an algorithm always retrieves all documents in a document base, it has one hundred per cent recall. However, it presumably has low precision. Typically recall and precision are plotted against each other; in the TREC evaluations, e.g., a "11-point" average measure is used, with precision measures at every 10 percent of recall, and the average figure is used as a total measure (e.g. Harman, 1996).

2.4 What is wrong with the evaluation measures?

As evaluation measures precision and recall have several very attractive qualities. They are intuitively valid and empirically can be determined to be reliable. However, they suffer from some distinctive draw-backs. For the required calculations the evaluator must

know how many relevant documents there are, how many documents there are in the base, how many documents are retrieved, how to weigh relevance to precision, how to determine what a query is, and how to judge relevance. All of these things can be done, but at risk of making the evaluation too ad-hoc.

Sampling

We do not have a good picture of how many documents are relevant in a given document base. Unless we have a small experimental database under complete experimental control we must resort to sampling procedures.

Universe

Indeed, often we cannot determine what the 'entire document base' is: for instance, in the case of Internet retrieval, where the database is fluid and huge by most contemporary standards.

Query

A query is well defined experimentally, but what its counterpart in real life is less well defined. Users often cannot pose their information needs in succinct search terms but cycle through a number of iterations until the visible top few items in a retrieved set seem satisfactory. At what point should we evaluate the system?

Retrieval

The retrieved set is not delimited in most probabilistic ranking systems. A list of several thousand documents is presumably not very useful to a human user. How many documents should we assume actually have been retrieved?

Precision vs recall

Averaging recall-precision trade offs in e.g. 11-point averages is common practice, but has the undesirable consequence to mask algorithm differences: some algorithms may do very well in high-precision searches and less well in high recall cases; some may do well in cases where there are very few documents to be found, others do better when the document base is saturated with material for the topic at hand.

For instance, morphological normalization of inflectional forms of a lexeme may raise recall. It may well lower precision concurrently. This may be desirable or not, depending on the task the user is engaged in.

Relevance

Most importantly: what is "relevant"?

Chapter 3

References

- Nicholas J. Belkin. 1994. "Design Principles for Electronic Textual Resources: Investigating Users and Uses of Scholarly Information", *Studies in the memory of Donald Walker*, Kluwer.
- Douglas Biber 1989. "A typology of English texts", *Linguistics*, 27:3-43.
- Benedict du Boulay, Tim O'Shea, and John Monk. 1981. "The Black Box Inside the Glass Box: Presenting Computing Concepts to Novices", *International Journal of Man-Machine Studies*, 14:237-249.
- Naoufel Ben Cheikh and Magnus Zackrisson. 1994. "Genrekategorisering av text för filtrering av elektroniska meddelanden" *Stockholm University Bachelor's thesis in Computer and Systems Sciences* Stockholm University.
- Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. "Indexing by Latent Semantic Analysis." *Journal of the American Society for Information Science*. 41(6):391-407.
- Marti Hearst and Christian Plaunt. 1993. "Subtopic Structuring for Full-length Document Access". *Procs. SIGIR '93*, Pittsburgh.
- Marti Hearst. 1994. "Multi-Paragraph Segmentation of Expository Text". *Procs. ACL '94*, Las Cruces.
- Jussi Karlgren. 1990. "An Algebra for Recommendations", *Syslab Working Paper* 179, Department of Computer and System Sciences, Stockholm University, Stockholm.
- Jussi Karlgren, Kristina Höök, Ann Lantz, Jacob Palme, and Daniel Pargman. 1994. "The Glass Box User Model for Filtering", *Procs. 4th International Conference on User Modeling*, Cape Cod. (Long version available as SICS Technical Report T94:09).
- Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergström, John Riedl. 1994. "GroupLens: An Open Architecture for Collaborative Filtering of Netnews", *Procs. CSCW 94*, Chapel Hill.
- Gerald Salton and Michael McGill. (1983). *Introduction to Modern Information Retrieval* New York: McGraw-Hill.
- Gerard Salton and James Allan. 1994. "Automatic Text Decomposition and Structuring", *Procs. 4th RIAO - Intelligent Multimedia Information Retrieval Systems and Management*, New York.

- Eva Hajičová, Hana Skoumalová, and Petr Sgall. 1995. An Automatic Procedure for Topic-Focus Identification. *Computational Linguistics* 21:1 81-95.
- Donald E. Walker. 1981. "The Organization and Use of Information: Contributions of Information Science, Computational Linguistics, and Artificial Intelligence", *Journal of the American Society for Information Science* **32**, (5), pp. 347-363.
- Ralph Grishman and John Sterling. 1990. Information Extraction and Semantic Constraints. *In Papers presented to the Thirteenth International Conference On Computational Linguistics (COLING -90)*, Hans Karlgren (ed.), Helsinki:University of Helsinki.
- Ralph Grishman. 1996. "The NYU system for MUC-6, or Where's the Syntax?". *Proceedings of the MUC workshop*. Washington.
- Ralph Grishman and Beth Sundheim. 1996. "Message Understanding Conference - 6: A Brief History". *In Papers presented to the Sixteenth International Conference On Computational Linguistics (COLING -96)*, Copenhagen: University of Copenhagen.
- Chris Jacquemin and Evelyn Tzoukermann "NLP for term variant extraction: Synergy between Morphology, Lexicon and Syntax" In Strzalkowski (1997).
- John S. Justeson and Slava M. Katz. 1995. "Technical Terminology: some linguistic properties and an algorithm for identification in text." *Natural Language Engineering*, 1:1 (9-27).
- Slava Katz. 1996. "Distribution of content words and phrases in text and language modelling." *Natural Language Engineering* 2:1:15-60.
- Kimmo Koskenniemi. 1996. "Finite state morphology in information retrieval." *Natural Language Engineering* 2:4.
- Hans Karlgren. 1976. *Homeosemy - On the Linguistics of Information Retrieval*. In Walker, Karlgren, and Kay (1976).
- Hans Karlgren. 1987. "Making Good Use of Poor Translations", in *International Forum On Information And Documentation*, 12:4, Moscow: FID.
- Hans Karlgren and Donald E Walker. 1980. The Polytext System - A New Design for a Text Retrieval System. In Kiefer (1980).
- Ferenc Kiefer (editor). 1980. Questions and Answers.
- Donald Kimber, Lynn Wilcox, Francine Chen, Thomas Moran. 1995. Speaker Segmentation for Browsing Recorded Audio. SIGCHI '95 (Denver Colorado, May 7-11, 1995) Human Factors in Computing System Conference Companion 1995. New York: ACM SIGCHI, pp. 212-213.
- Wessel Kraaij and Renée Pohlmann. 1996. "Viewing Stemming as Recall Enhancement". *Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval*. Zürich. ACM: New York.
- Gunnel Källgren. 1979. *Innehåll i text*. Ord och Stil 11. Lund: Studentlitteratur.
- Gunnel Källgren. 1978. "Deep Case, Text Surface, and Information Structure". *Nordic Journal of Linguistics* 1:149-167.
- Hans Peter Luhn. 1957. "A Statistical Approach to Mechanical Encoding and searching of Literary Information". *IBM Journal of Research and Development* 1:4 (309-317).

- Hans Peter Luhn. 1958. "The Automatic Creation of Literature Abstracts". *IBM Journal of Research and Development* 2:2 (159-165; 317).
- Hans Peter Luhn. 1959. "Auto-Encoding of Documents for Information Retrieval Systems". In M. Boaz (editor). *Modern Trends in Documentation*. London: Pergamon Press. (45-58).
- S. E. Robertson and Karen Sparck Jones. 1996. Simple, proven approaches to text-retrieval. Technical report 356, Computer Laboratory, University of Cambridge. [<http://www.cl.cam.ac.uk/ftp/papers/reports/TR356-ksj-approaches-to-text-retrieval.ps.gz>]
- M. F. Porter. 1980. "An algorithm for suffix stripping". *Program* 14:3 (130-137).
- Petr Sgall. 1980. Relevance of Topic and Focus for Automatic Question Answering. In Kiefer (1980).
- Karen Sparck Jones and Martin Kay. 1971. Linguistics and Information Science. Report to the FID/LD.
- Karen Sparck Jones and Martin Kay. 1976. Linguistics and Information Science: A Postscript. In Walker, Karlgren, and Kay (1976).
- Karen Sparck Jones. 1997. "What is the role of NLP in Text Retrieval?". In Strzalkowski (1997).
- Tomek Strzalkowski. 1994. "Robust Text Processing in Automated Information Retrieval". *Proceedings of the Fourth Conference on Applied Natural Language Processing in Stuttgart*. ACL.
- Tomek Strzalkowski. 1994. "Building a Lexical Domain Map from Text Corpora." In *Papers presented to the Fifteenth International Conference On Computational Linguistics (COLING-94)*, Kyoto.
- Tomek Strzalkowski (editor). 1997. *Natural Language Information Retrieval*. Boston: Kluwer.
- Tomek Strzalkowski, Louise Guthrie, Jussi Karlgren, Jim Leistsnyder, Fang Lin, José Pérez-Carballo, Troy Straszheim, Jin Wang, Jon Wilding. 1997. "Natural Language Information Retrieval: TREC-5 Report" *Proceedings of the fifth Text Retrieval Conference, TREC-5*. Donna Harman (ed.), NIST Special Publication, Gaithersburg: NIST.
- Tomek Strzalkowski and Jin Wang. 1996. A Self-Learning Universal Concept Spotter. In *Papers presented to the Sixteenth International Conference On Computational Linguistics (COLING-96)*, Copenhagen.
- Donald E. Walker, Hans Karlgren, and Martin Kay (editors). 1976. Natural Language in Information Retrieval - Perspectives and Directions for Research. Results of a workshop on linguistics and information science. Biskops-Arnö, Sweden, May 1976, organized by the Committee on Linguistics in Documentation of the International Federation of Documentation and KVAL. FID publication 55. Stockholm: Skriptor.
- Jussi Karlgren, Kristina Höök, Ann Lantz, Jacob Palme, Daniel Pargman. (1994). "The glass box user model for filtering." Submitted to the First International Conference On User Modeling, Cape Cod.
- Erik Andersson. 1975. "Style, optional rules and contextual conditioning. In *Style and Text - Studies presented to Nils Erik Enkvist*. Håkan Ringbom. (ed.) Stockholm: Skriptor and Turku: Åbo Akademi.

- Douglas Biber. 1988. *Variation across speech and writing*. Cambridge University Press.
- Douglas Biber. 1989. "A typology of English texts", *Linguistics*, 27:3-43.
- Chris Buckley, Amit Singhal, Mandar Mitra, Gerard Salton. 1996. "New Retrieval Approches Using SMART: TREC 4". *Proceedings of TREC-4*.
- Naoufel Ben Cheikh and Magnus Zackrisson. 1994. "Genrekategorisering av text för filtrering av elektroniska meddelanden" (Genre Classification of Texts for Filtering of Electronic Messages) *Stockholm University Bachelor's thesis in Computer and Systems Sciences*, Stockholm University.
- John Dawkins. 1975. *Syntax and Readability*. Newark, Delaware: International Reading Association.
- Nils Erik Enkvist. 1973. *Linguistic Stylistics*. The Hague: Mouton.
- Donna Harman (ed.). 1995. *The Third Text REtrieval Conference (TREC-3)*. National Institute of Standards Special Publication. Washington.
- Donna Harman (ed.). 1996. *The Fourth Text REtrieval Conference (TREC-4)*. National Institute of Standards Special Publication 500-236. Washington.
- Donna Harman (ed.). forthcoming. *The Fifth Text REtrieval Conference (TREC-5)*. National Institute of Standards Special Publication. Washington.
- Marti Hearst. 1997. "TextTiling: Segmenting Text into Multi-Paragraph Subtopic Passages". *Computational Linguistics*: to appear March 1997.
- Marti Hearst and Christian Plaunt. 1993. "Subtopic Structuring for Full-length Document Access". *Proceedings of the 16th ACM SIGIR Conference on Research and Development in Information Retrieval*. Pittsburgh. New York: ACM.
- Marti Hearst. 1994. *Context And Structure In Automated Full-Text Information Access*. Ph D. Thesis. University of California, Berkeley.
- Fahima Polly Hussain and Ioannis Tzikas. 1995. "Ordstatistisk kategorisering av text för filtrering av elektroniska meddelanden" (Genre Classification of Texts by Word Occurrence Statistics for Filtering of Electronic Messages) *Stockholm University Bachelor's thesis in Computer and Systems Sciences*, Stockholm University.
- Jussi Karlgren. 1996. "Stylistic Variation in an Information Retrieval Experiment" In Proceedings NeMLaP 2, Bilkent, September 1996. Ankara: Bilkent University. (In the Computation and Language E-Print Archive: cmp-lg/9608003).
- Jussi Karlgren and Douglass Cutting. 1994. "Recognizing Text Genres with Simple Metrics Using Discriminant Analysis", Proceedings of COLING 94, Kyoto. (In the Computation and Language E-Print Archive: cmp-lg/9410008).
- Jussi Karlgren and Troy Straszheim. 1997. "Visualizing Stylistic Variation." *In the Proceedings of the 30th HICSS*, Maui.
- George R. Klare 1963. *The Measurement of Readability*. Iowa Univ press.
- Irving Lorge. 1959. *The Lorge Formula for Estimating Difficulty of Reading Materials*. New York: Teachers College Press, Columbia University.

- Robert M. Losee. forthcoming. "Text Windows and Phrases Differing by Discipline, Location in Document, and Syntactic Structure". *Information Processing and Management*. (In the Computation and Language E-Print Archive: cmp-lg/9602003).
- Mendenhall, T.C. 1887. "The Characteristic Curves of Composition." *Science* 9: 237-49.
- I. I. Menshikov. 1974. "K voprosu o zhanrovo-stilevoy obuslovlennosti sintaksicheskoy struktury frazy". ("On genre-dependent stylistic variation of the syntactic structure in the clause") In *Voprosy statisticheskoy stilistiki*. Golovin et al. (eds.) 1974. Kiev: Naukova dumka; Akademia Nauk Ukrainy SSR.
- J. Ross Quinlan. 1993. *C4.5: Programs for Machine Learning*. San Mateo: Morgan Kaufmann.
- Rolf Sandell. 1977. *Linguistic Style and Persuasion*. European Monographs in Social Psychology 11. London: Academic Press.
- Josef Vachek. 1975. "Some remarks on functional dialects of standard languages". In *Style and Text - Studies presented to Nils Erik Enkvist*. Håkan Ringbom. (ed.) Stockholm: Skriptor and Turku: Åbo Akademi.
- Ellen Voorhees, Narendra K. Gupta, Ben Johnson-Laird. 1994. "The Collection Fusion Problem". *Proceedings of TREC-3*.
- Brian T. Bartell, Garrison W. Cottrell, and Richard K. Belew. 19xx. Automatic Combination of Multiple Ranked Retrieval Systems. xxx...xxx.
- Douglass R. Cutting, David R. Karger, Jan O. Pedersen, John W. Tukey. 1992. Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections. Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Copenhagen. New York: ACM.
- John S. Justeson and Slava M. Katz. 1995. Technical Terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1, 1, 9-27.
- Jussi Karlgren and Kristofer Franzń. 1997. Verbosity and Interface Design. Reptile working papers No. 2. [<http://www.sics.se/jussi/irinterface.html>].
- Magnus Merkel, Bernt Nilsson and Lars Ahrenberg. 1994. A Phrase-Retrieval System Based on Recurrence. In Proceedings from the Second Annual Workshop on Very Large Corpora. Kyoto.
- Rebecca J. Passonneau and Diane J. Litman. 1993. "Intention-based Segmentation: Human Reliability and Correlation with Linguistic Cues" *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*. ACL.
- Jürgen Koenemann, Nicholas J. Belkin. 1996. A Case For Interaction: A Study of Interactive Information Retrieval Behavior and Effectiveness. SIGCHI '96 (Vancouver, April 14-18, 1996) Human Factors in Computing System 1996. New York: ACM SIGCHI, pp. 205-212.
- S. E. Robertson and Karen Sparck Jones. 1996. Simple, proven approaches to text-retrieval. Technical report 356, Computer Laboratory, University of Cambridge. [<http://www.cl.cam.ac.uk/ftp/papers/reports/TR356-ksj-approaches-to-text-retrieval.ps.gz>]

- Daniel E. Rose and Curt Stevens. 1996. V-Twin: A Lightweight Engine for Interactive Use. Proceedings of the fifth Text Retrieval Conference, TREC-5. Donna Harman (ed), NIST Special Publication, Gaithersburg: NIST.
- Gerard Salton and Christopher Buckley. 1988. Term-Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management*. 24 (5) 513-523.
- Gerard Salton and C. S. Yang. 1973. On the Specification of Term Values in Automatic Indexing. *The Journal of Documentation*. 29 (4) 351 - 372.
- Amit Singhal, Gerard Salton, Mandar Mitra, Chris Buckley. 19xx. Document Length Normalization. Cornell CS TR000.
- Frank Smadja. 1993. Retrieving Collocations from Text: XTRACT. *Journal of Computational Linguistics*. Special issue on corpus based techniques. 19 (1) 143 - 177.
- Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*. December 1972. 28:1:11-20.
- Takenobu Tokunaga and Makoto Iwayama. 1994. Text categorization based on weighted inverse document frequency. Technical Report 94 TR0001. Department of Computer Science. Tokyo Institute of Technology.