Linguistically Motivated Information Retrieval[*]

Avi Arampatzis[†] Th.P. van der Weide P. van Bommel C.H.A. Koster

September 1999
Revised for publication on September 2000

To appear in:
Encyclopedia of Library and Information Science,
Volume 69, December 2000.
Allen Kent, editor.
Published by Marcel Dekker, Inc., New York, Basel.

Keywords: Information Retrieval, Natural Language Processing, Phrase Retrieval Hypothesis, Linguistic Normalization.

[*]Parts of this article are adapted from Information Processing and Management, Vol 34, No 6, "Phrase-based Information Retrieval", pp 693-707, 1998, with permission from Elsevier Science.

[†]Dept. of Information Systems, Faculty of Mathematics and Computing Science, University of Nijmegen, Toernooiveld 1, 6525 ED Nijmegen, The Netherlands. Fax: +31 24 3553450

Contents

1 Introduction

Information retrieval (IR) has been developed to provide practical solutions to people's need to find the desired information in large collections of data. The IR task can be seen as the "digital twin" of the task of a person looking in a library for material relevant to a certain subject. In both cases, the searcher has an *information need* that has to be translated to library indices or *query* terms. Then it is submitted to some system — library catalogue or computerized retrieval system — and the system in turn suggests (retrieves) relevant material. The searcher will usually find that some of the suggested documents are not actually relevant, and will also suspect that some relevant documents might have been missed. For static collections, the effectiveness of such a search can be quantified using two metrics, *precision* and *recall*. Precision is defined as the ratio of the number of relevant retrieved documents to the total number of retrieved documents. Recall is the ratio of the number of relevant retrieved documents to the total number of relevant documents in the document collection. For an extended introduction to the IR problem, its history, widely accepted techniques, and retrieval evaluation metrics, the reader should refer to the classical books [1] and [2]; for a collection of classical articles in IR, to [3] (all in Readings for Further Study).

The tremendous increase over the last decade in information in digital form has led to a new challenge in IR. A World Wide Web search today involves large amounts of information, and going through hundreds of irrelevant hits, which is usually the case, is very tedious. Although IR has been in existence for more than three decades (and as a part of library science for much more), modern technology for its part is still based on a simple assumption that often leads to unsatisfactory results. Restricting the problem to textual data, the assumption, implicit or explicit, upon which most commercial IR systems are based, is that

Definition 1 (naive keyword retrieval hypothesis) *If a query and a document have a (key)word in common, then the document is to some extent about the query.*

Of course, if they have *more* keywords in common, then the document is *more* about the query. Moreover, the keywords are usually augmented with weights indicating their importance as information discriminators. In that respect, the IR problem is represented by matching the "bag" of keywords in the user's query with the bag of keywords representing the documents. The output of such a matching is usually a *ranked* list of documents with the most relevant first and the least relevant last.

This relatively simple representation is the computer-age equivalent of library catalogues, and carries the same inadequacies. The most obvious inadequacies originate from *linguistic variation*, making the keyword retrieval hypothesis insufficient because

1. It does not deal with *morphological variation* which produces keywords in different numbers, for instance *wolf* and *wolves*, or different cases, such as *man* and *man's*. (Dealing with cases is trivial for English, but it is crucial for other more inflected languages like German or Greek).

2. It does not handle cases in which different words are used to represent the same meaning. For this phenomenon we use the term *lexical variation*. The result is that a query with the keyword *film* does not retrieve documents that contain its synonym, *movie*.

3. It does not distinguish cases in which single words have multiple meanings due to *semantical variation*. A singer looking for *bands* will be faced with *radio frequency*

bands as well.

4. It does not deal sufficiently with *syntactical variation.* A document that contains the phrase *near to the river, air pollution is a major problem* is not about *river pollution*, although both keywords occur in the document, and certainly *science library* is not the same as *library science.*

Linguistic variation degrades the effectiveness of IR systems in terms of precision and recall. On the one hand, morphological and lexical variation hurts recall. On the other hand, semantical and syntactical variation hurts precision. However, trying to improve recall usually decreases precision and vice versa.

Linguistic variation in the IR context may be interpreted as meaning that language is not merely a bag of words. Language is a mean to communicate about concepts, entities, and relations, which may be expressed in many forms. Word order may matter (as in *science library* vs. *library science*) or may not (*general director* vs. *director general*). Moreover, words combine to form phrases and other larger units with a meaning that may not be directly inheritable from the individual words. For example, a *hot dog*, either hot or not, has nothing to do with dogs. Given such considerations, it has been conjectured that a better representation should also include groups of words (phrases) and some form of regularization of words, word order, and meaning. Indeed, many researchers have developed such techniques.

This article discusses a retrieval schema that attempts to overcome the problems originating from the keyword retrieval hypothesis and linguistic variation. It is partly based on [1], and is organized as follows. In the next section we will review some of the most important attempts made to deal with linguistic variation. In the rest of this article we will discuss the key aspects of a linguistically motivated retrieval system. Starting in section 3 from a phrase retrieval hypothesis — a naive extension of the keyword retrieval hypothesis — we will address a suitable for IR representation of phrases in section 4. In section 5, possible regularizations of natural language will be outlined. The weighting of phrasal indexing terms and their matching will be discussed in section 6. An example architecture of such a linguistically motivated retrieval system will be depicted in section 7. We will draw some conclusions in the final section.

2 Dealing with Linguistic Variation

The problems of linguistic variation have been noted by many researchers, who have answered with various techniques. Many of these techniques employ natural language processing (NLP) and such language resources as online dictionaries and thesauri. The results until now have been inconsistent, making it difficult to reach a conclusion about their effectiveness. We will review some approaches and their outcomes for each of the morphological, lexical, semantical, and syntactical variation.

2.1 Morphological Variation

Morphology is the area of linguistics concerned with the internal structure of words. It is usually broken down to two types, *inflectional* and *derivational.* Inflectional morphology describes the predictable changes a word undergoes as a result of syntax, and has no effect on the word's part of speech (e.g., a noun remains a noun) and little effect on its meaning. The most common changes are the plural and possessive forms of nouns (e.g., *computer, computers, computer's*), comparative and superlative form of adjectives (e.g., *good, better, best*), and the past tense, past participle, and progressive form of verbs (e.g., *compute, computed, computing*). On the contrary, derivational morphology may or may not affect part of speech or meaning (e.g., *computerize, computerization*).

Two ways have generally been followed to deal with morphology in IR trying to increase recall. These are *query expansion* and *stemming.* In query expansion, morphological variants of keywords are added to the query. Stemming simply strips a word's suffix to reduce it to its *stem*, assuming that keywords with a common stem usually have similar meanings. Query expansion and stemming can been regarded as equivalent and the choice depends on the nature of the particular application. We will concentrate on stemming as the choice that is made the most.

Stemming can be done in a linguistic fashion, taking into account the function and the part of speech of a word, or in a nonlinguistic fashion, disregarding a word's context.

Lovins and Porter developed nonlinguistic algorithms for suffix stripping based on a list of frequent suffixes to reduce words to their stems [2, 3]. It is a common belief that stemmers improve recall without losing too much precision, however, a comparison of the Lovins stemmer, the S stemmer, and the Porter stemmer with a baseline of no stemming at all, concluded after detailed evaluation that none of the three stemming algorithms consistently improves retrieval for English documents [4]. It was argued that the evaluation measures were not appropriate, and new measures were proposed for evaluating the performance of different stemming algorithms [5]. After experimentation, it was concluded that stemming is almost always beneficial for English, except for long queries at low recall levels. A more reliable version of Porter's stemmer was developed, which uses a dictionary to validate the result after every suffix-stripping step. This revised Porter stemmer resulted in improvements in retrieval performance for English documents, especially short ones [6].

Research with other morphologically more complex languages such as Slovene showed an improvement in effectiveness using a Porter-like stemmer modified for Slovene [7]. In the same study, when the Slovene corpus was translated to English and the experiment was repeated, there was no improvement in retrieval. For Dutch texts, it was found that linguistic inflectional stemming improves recall without significant loss in precision, while derivational stemming, although sometimes useful, in general reduces precision too much [8].

2.2 Lexical Variation

Lexical variation has generally been treated in two ways. On the one hand, by (lexical) *query expansion* with semantically related terms (e.g., synonyms), and on the other hand, the matching of query and document keywords via *conceptual distance measures.* For these purposes, thesauri have been exploited to supply related query terms, and semantical networks such as this of WordNet [9] to define semantical distance measures between words.

The choice of semantically related terms for a word depends on the context in which the word is used; thus, the context specifies the word's *sense.* When a word can be used in different senses, the problem of *word sense ambiguity* arises. Most of the techniques that deal with lexical variation require prior word sense disambiguation, and that makes these techniques strongly dependent on semantical variation (described in the next section).

Query expansion with WordNet has shown a potential in enhancing recall since it permits the matching of relevant documents that do not contain any of the query terms [10].

Expansion of queries using synonymy and other semantic relations supported by WordNet showed that short and incomplete queries can be significantly improved, yielding better retrieval effectiveness [11]. However, this query expansion technique made little difference in the effectiveness, for relatively complete descriptions of the information sought. For Dutch texts, synonym expansion was reported as potentially useful [12].

Experiments on a small collection of image captions (i.e., very short documents) using measures of semantical similarity distance between words based on WordNet showed improvements in retrieval [13]. However, their earlier experiments with word-to-word semantical similarity measures resulted in a drop in effectiveness, due to the effects of erroneous word sense disambiguation [14].

Another approach, based on indexing in terms of WordNet's synonym sets (synsets) instead of wordforms, yielded successful results when queries were fully disambiguated [15]. If queries are not disambiguated, indexing by synsets at best performs only as well as standard word indexing.

2.3 Semantical Variation

Semantical variation has strong impacts on lexical query expansion, on matching based on word-to-word semantical distance similarity measures, and on conceptual indexing. The success of these techniques requires prior disambiguation of word senses, as many researchers have noted [11, 12, 13, 15]. Most of the research has concentrated on how large the impact of semantical variation and its inaccurate resolution is on IR effectiveness.

It is estimated that if word sense disambiguation is performed with less than 90% accuracy the retrieval results are worse than not disambiguating at all [16]. Poor retrieval results were blamed on this reason in previous research [14]. Conversely, in the same experiments [16] word sense ambiguity was shown to produce only minor effects on retrieval accuracy, apparently suggesting that query-document matching strategies already perform an implicit disambiguation. In this experimental setup, ambiguity was introduced artificially by substituting randomly selected word pairs such as *bank* and *spring* with ambiguous terms like *bank/spring*. This setup has two disadvantages, first, real ambiguity might not behave like the artificially introduced one, and second, the disambiguation of an artificially ambiguous term is only partial; when *bank/spring* is disambiguated as *bank*, *bank* is still ambiguous as it can be used in more than one sense in a text collection [15].

2.4 Syntactical Variation

The techniques developed to deal with syntactical variation may be grouped in two categories: the addition of *phrases* to queries, and the use of *syntactical structures* for indexing. These techniques intend to increase retrieval precision.

A phrase is a group of words, and historically what has been referred to as a phrase in the IR context varies significantly among researchers. The hypothesis for using phrases has been that they denote more meaningful entities or concepts than single words; thus they may constitute a better representation. Indeed, the use of phrases has become common in IR; many systems participating in the text retrieval conferences (TRECs) now use one or another form of phrase extraction [17].

Traditionally, two types of phrases have been used, *statistical* and *syntactic*. Statistical phrases are any series of words that frequently occur contiguously in a text collection.

Syntactic phrases are any set of words that satisfy certain syntactic relations or constitute specified syntactic structures. Statistical phrases are extracted using word frequency and co-occurrence information, while syntactic phrases usually require sophisticated NLP techniques. Which of the two types is more useful for IR remains unclear; syntactic phrases seem to offer an advantage that is statistically rather insignificant [18, 19, 20].

The addition of syntactic phrases to queries yielded a substantial improvement in precision, especially near the top of the ranking [21]. This benefit, however, was tied to the length of the query: the longer the query, the larger the improvement. Significant improvements in retrieval performance were found when syntactic phrases supplemented single words [22]. However, the impact of adding phrases varied according to the query topic. Adding phrases helped some topics, while it hurt some others. Small statistically insignificant improvements were also found for Dutch texts [20]. Other research concluded that phrases do not have a major effect in precision at high ranks, but are more useful at lower ranks [19].

Lexical atoms, such as *hot dog*, were used to *replace* their single words in indexing [22]. The experiments with replacing high-frequency adjacent word-pairs — only adjective–noun and noun–noun combinations — with the corresponding phrase for indexing showed an improvement in the average precision. Nevertheless, the inconsistent influence of phrases on recall and initial precision suggested a need for a better control over the selection of phrases that are used for replacing single words.

Indexing structures derived from syntax were tried in [23]. The matching between queries and documents was based on tree structures constructed from clauses. Syntactic ambiguity was also encoded in these tree structures and taken into account by weighting various syntactic interpretations at the time of retrieval. The experimental results were disappointing in both precision and recall. The group gave as possible reasons for the poor results the poor quality of the language analyzer, the different type of language in documents and queries, and the retrieval strategy applied.

2.5 Beyond the Bag-of-words Paradigm

Various attempts have been made to break out of the bag-of-words paradigm. Experiments have shown considerable variation in retrieval effectiveness, making it difficult to establish which techniques actually work and which do not. Summarizing

- The effectiveness of stemming is dependent on the morphological complexity of a language. Restricting the problem to English, there is a lot of variation in the results of stemming experiments, and a number of factors seem to be of importance, e.g., linguistic vs. nonlinguistic stemming, stemming algorithm, query and document length, and even evaluation measures.

- Lexical and semantical variation are strongly connected. It seems that dealing with lexical variation is more beneficial for incomplete and relatively short queries. It is still an unanswered question if conceptual distance matching scales up to longer documents and queries. Moreover, most of the relevant research has shown that the successful application of these techniques is very sensitive to word-sense ambiguity. However, word sense disambiguation techniques are still not well established.

- It is still not clear how syntactical information can be used to improve retrieval effectiveness consistently. Questions still remain about which phrases are useful, in which cases, and how these should be extracted. Furthermore, NLP is still nowhere near to becoming practical in dealing with large amounts of textual data of unrestricted domain. Due to its lack of robustness and efficiency, compromises have to be made. NLP techniques have mostly been used to *add* indexing terms in a bag-of-words representation, and therefore trying to sharpen a keyword-based search. In that way, the inadequacies of NLP have been softened; in the worst case a system will fall back to the original bag-of-words representation.

Although a lot of effort has been put into linguistically motivated retrieval schemes, whether or not this is worth the trouble remains unclear. The evidence suggests the need for further investigation and better modeling. In the rest of this study, we will describe a retrieval scheme that demonstrates the application of linguistically motivated techniques.

3 The Phrase Retrieval Hypothesis

The goal of the *indexing* task is to assign characterizations (*terms*) to documents that are deemed to best represent their content. Every term used to characterize documents of the same collection can be seen as adding a new dimensionality to the characterization. Terms should be assigned to documents in such a way that documents on the same topic are positioned close together in the N-dimensional term space, while those on different topics are placed sufficiently apart. Terms can be anything from, for example, tri-grams and words to linguistic-entities and concepts. In the two extreme cases, documents can be characterized by themselves (e.g., their document numbers), or all documents by exactly the same characterization. The former characterization positions documents as far apart as possible, resulting in no way of retrieving documents on the same topic. It is thus unusable in the IR context. The latter provides no way of discriminating between different topics, therefore a suitable characterization must be *usable* and *discriminating*.

In a keyword-based representation, every document is characterized by a set of keywords with weights representing the importance of each keyword in characterizing the document. Keywords are usually derived directly from the document's text. Keyword-based representations are modestly usable and discriminating. Single words are rarely specific enough for accurate representation (e.g., the word *system* does not say much, whereas a *sound system* clarifies the meaning somewhat more). Moreover, a word with a high frequency of occurrence in a document collection is not a good discriminator. On the other hand, a phrase, even made up of high-frequency words, may occur in only a few documents, thus becoming a good discriminator. These observations suggest that a better characterization will make use of phrases; consequently, a *naive* phrase retrieval hypothesis can be formalized as follows:

Definition 2 (naive Phrase Retrieval Hypothesis) *If a query and a document have a phrase in common, then the document is to some extent about the query.*

The phrase retrieval hypothesis does not solve the problems originating from the keyword retrieval hypothesis and linguistic variation. On the contrary, it creates more questions, such as what a phrase is and how it should be used for indexing or be weighted and matched. We use this definition merely as a starting point, upon which we will build our framework.

Phrases can be obtained using statistical or syntactic methods. Syntactic phrases appear to be reasonable indicators of content, arguably better than proximity-based statistical phrases, since they account for word-order changes or other structural constructions (e.g., *science library* vs. *library science* vs. *library of science*). Experiments have shown, however, that syntactic methods are not significantly more effective than statistical methods [18, 19, 20]. This failure of NLP to outperform statistics can be attributed to the poor quality and robustness of the existing NLP techniques. Nevertheless, we will adopt a syntactic approach for the time being, assuming that accurate syntactic analysis and disambiguation techniques will become available. We will return to the effectiveness issues of NLP in section 7.

Evidence suggests that noun phrases should be considered as a semantical unit. The most important reasons are

- noun phrases play a central role in the syntactic description of *all* natural languages, functioning as subject and object, and in preposition phrases.
- In artificial intelligence, noun phrases are considered as references to (or descriptions of) complicated concepts [24]. By others, as *picture producers.*

Noun phrases might be good approximations of concepts, but other phrases also corresponding to concepts are missed. This observation points to the necessity to consider other phrases as well (e.g., verb phrases). The verb phrase describes a situation or process by relating a main verb to a number of noun phrases and other phrases. The linguistically meaningful phrases that may be considered as retrieval terms are therefore at least the noun phrase including its modifiers, and the verb phrase including its subject, object, and other complements. An abstract representation of these phrases suitable for indexing is needed, and will be defined in section 4.

Phrases can be used in their literal form as terms, although the performance is then expected to be inferior to that of keywords. It is well known that as the size of a corpus grows, the number of keywords grows with the square root of the size of the corpus. One could expect that the same holds for phrases, but the number of such enriched terms grows even faster, as does the likelihood of there being different phrases corresponding to the same concept. On the one hand, we would like to use phrases to achieve precision, but on the other hand, recall will be too low because the probability of a phrase reoccurring literally is too low. To deal with this *sparsity* of phrasal terms, we shall introduce a number of *linguistic normalizations* (section 5). Linguistic normalization tries to reduce alternative formulations of meaning to a *normalized form.* For example, *river pollution* and *pollution of rivers* are both normalized to the same indexing term `pollution+river`.

4 Representation of Phrases

A syntactic phrase can be represented in various ways. At the bottom end of the representation spectrum, a phrase can be represented simply by the unordered set of its words, disregarding *all* structure. At the other end, all linguistic structure can be taken into account, resulting in complicated parse-tree representations. The choice is a trade-off between syntactic information and the ease of phrase extraction.

For example, a simple noun phrase picker could easily be constructed by looking for sequences of articles, adjectives, and nouns within a text. A noun phrase extracted like that would contain little information about how its adjectives and nouns are related to each other, except that adjacent words are most probably more related than nonadjacent ones. In an unordered set-of-words representation, and assuming there is no special treatment of proper names, the noun phrase

the hillary clinton health care bill proposal

would contain *bill clinton*, but it is obvious that this phrase does not refer to him. However, experimentally such a co-occurrence of query keywords within a noun phrase has resulted in clear improvements in precision [25]. A sequence-of-words representation does not contain *bill clinton* (rightly), but does not contain *clinton proposal* either (wrongly). A full linguistic parsing would result in a much more precise representation. The parse-tree would contain too much linguistic detail, however, most of which is unnecessary for indexing, as such details reflect mostly the syntactic description of the natural language used rather than the intended meaning. Since the goal is to derive adequately precise (for retrieval purposes) meaning from syntax, we will settle for less than full linguistic parsing. Linguistically motivated *light* parsing has already been shown to slightly improve retrieval results over the classic IR approximation to noun phrase recognition [26].

As a result, an intermediate representation of noun and verb phrases is desirable, eliminating structures that can be assumed not to be beneficial to IR:

Definition 3 (noun phrase for IR) *A core noun phrase NP, from an IR point of view, has the general form:*

$$NP \ = det^{*} pre^{*} head\ post^{*}$$

where

- *det (determiner) = article, quantor, number, etc.*
- *pre (premodifier) = adjective, noun, or coordinated phrase.*
- *head = usually a noun.*
- *post (postmodifier) = prepositional phrase, relative clause, etc.*
- *the asterisk ($*$) denotes a list of zero or more elements.*

Pre- and postmodifiers may recursively include other NPs.

Definition 4 (verb phrase for IR) *A verb phrase VP, from an IR point of view, has the general form:*

$$VP \ = subj\ kernel\ comp^{*}$$

where

- *subj (subject) = an NP (in the wide sense, including personal names and personal pronouns).*
- *kernel (verbal clause) = inflected form of some verb, possibly composed with other auxiliary verbforms and adverbs.*
- *comp (complements, such as object, indirect object, or preposition complement) = an NP or prepositional phrase (PP).*

- *the asterisk (∗) denotes a list of zero or more elements, depending on the transitivity of the verb (e.g., intransitive verbs have no complements, transitive verbs have an object, ditransitive have an object and indirect object).*

In accordance with the above definitions, it is possible to perform a parsing arguably lighter than full linguistic parsing, while a reasonable amount of structural information will still be retained. An example parsetree is given in figure 1. This is rather compact in comparison with a full linguistic parsetree, which would easily have overrun this page for the same sentence. Of course it is important that the parser is able to deduce the correct (or at least the most

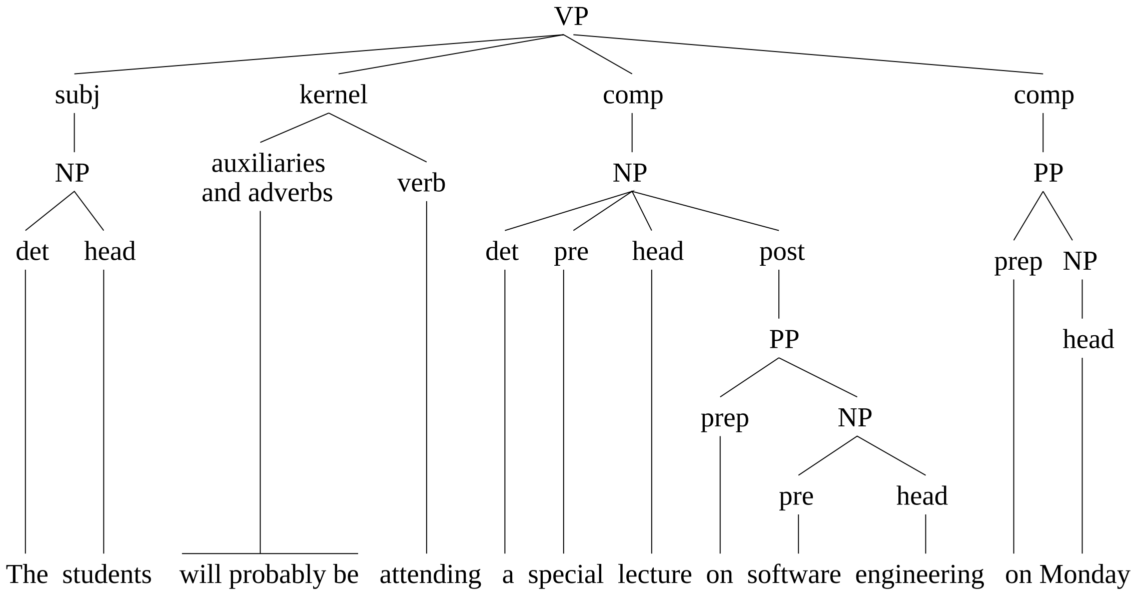

Figure 1: light parsing for IR purposes.

probable) dependency structure in complicated phrases. As we will see next, some elements that are considered of little interest from an IR point of view (e.g., determiners, prepositions, auxiliaries, and adverbs), may be eliminated.

5 Linguistic Normalization

The goal of normalization is to map different but semantically equivalent phrases onto one canonical representative phrase, the *phrase frame* (Figure 2). We distinguish between three types of normalization: the *morphological, syntactical*, and *lexicosemantical* normalization.

5.1 Morphological Normalization

Morphological normalization has traditionally been performed by means of stemming. Non-linguistic stemming, especially when it operates in the absence of any lexicon at all, is rather aggressive and may result in improper conflations. For instance, a Porter-like stemmer without a lexicon will reduce *university* to *universe* and *organization* to *organ*. Errors such as these are translated into a loss in retrieval precision. This impact is greater for more inflected languages than English because of the increased number of introduced ambiguities. Such

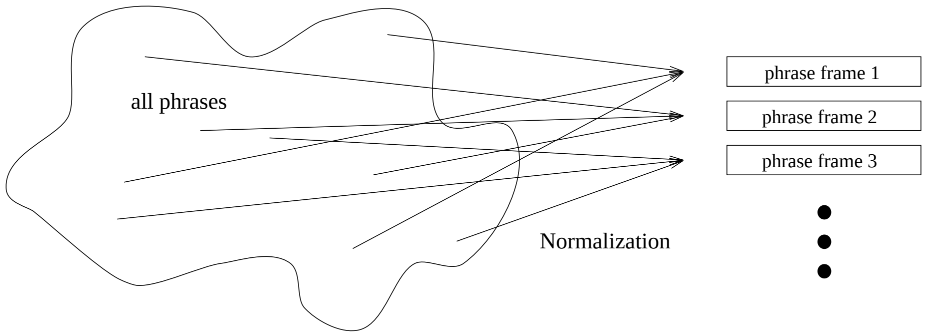

Figure 2: linguistic normalization

improper conflations can be avoided by simply checking for the existence of the wordform in a lexicon after each reduction step. Nevertheless, the verb forms *attached* and *suited* will still be reduced wrongly to the nouns *attaché* and *suite*, respectively.

Taking into account the linguistic context, a more conservative approach will prevent many of these errors. Conflations can be restricted to retain the part of speech of a word. In this respect, morphological normalization may be performed by means of *lemmatization*.

1. Verb forms are reduced to the infinitive.

2. Inflected forms of nouns are reduced to the nominative singular.

3. Comparatives and superlatives of gradable adjectives are reduced to the absolute form.

For this task, the grammatical rules for forming, for example, past participles or noun plurals, should be applied in a reverse way. Furthermore, the utilization of exception lists in order to handle irregularities such as *wolf–wolves*, *bad–worse–worst*, and *see-saw-seen* is indispensable.

Lemmatization is relatively simple and handles mostly inflectional morphology. It is similar to the *lexicon-based word normalization*, as referred to in [27]. It must be noted that there are cases in which lemmatization reduces noun and verb forms to the same lemma. Consider, for instance, the verb form *attacked* and the plural noun *attacks*; both will be lemmatized as *attack*. Although such conflations seem beneficial, there are indications from a text filtering experiment that the confusion between nouns and verbs when these are lemmatized, decreases the effectiveness [28].

Derivational morphology involves semantics and cross part-of-speech word relations, and hence should be approached carefully. Certain derivational transformations may be suggested by syntax. For instance, verbs may be turned into nouns (nominalization) or the other way around, as will be shown in the next section. The remaining derivational morphology should be treated where possible by lexico-semantical normalization.

5.2 Syntactical Normalization

According to the linguistic principle of *headedness*, any phrase has a single head. This head is usually a noun (the last noun before the postmodifiers) in NPs, and the main verb in the case of VPs. The rest of the phrase consists of modifiers. Consequently, every phrase can be mapped onto a *phrase frame*

$$PF = [h, m]$$

The head h gives the central concept of the phrase and the list m of modifiers serves to make it more precise. Conversely, the head may be used as an abstraction of the phrase, losing precision but gaining recall. It should be noted that although the head–modifier relation implies semantic dependence, what we have here is purely a syntactic relation. The intention is to produce meaningful indexing terms without deep semantic analysis, therefore the precise semantic interpretation of any head–modifier relation is forborne, treating it simply as an ordered relation.

Heads and modifiers in the form of phrases are recursively defined as phrase frames: $[[h_1, m_1], [h_2, m_2]]$. The modifier part may be empty in case of a bare head. This case is denoted equivalently by $[h,]$ or $[h]$. The head may serve as an index for a list of phrases with occurrence frequencies

```
[ engineering   1026 ,
      of software 7 ;
      reverse   102 ;
      software  842 ;
      ... ]
```

where the frequency of a bare head includes that of its modified occurrences. Alternative modifications of the head are separated by semicolons.

Phrases frames are produced by normalizing the phrase representations of definitions 3 and 4. In noun phrases, determiners are of little interest for IR, thus they may be eliminated. The normalization of noun phrase is defined as

Definition 5 (noun phrase normalization)

$$NP \;= det^* pre^* head\ post^* \mapsto [head, pre^* post^*]$$

The elements of the list $pre^* post^*$ are considered to modify the head *independently* from each other, and they are separated by semicolons, hence any PF containing a list, e.g., $[h, m] = [h, m_1; m_2]$, may be expanded as $[h, m_1]; [h, m_2]$. The noun phrase normalization can be applied recursively on heads and modifiers that include other NPs. For example

$$\begin{aligned} &\textit{a special lecture on software engineering} \mapsto \\ \mapsto\ &\texttt{[lecture, special; on software engineering]} \mapsto \\ &\mapsto \texttt{[lecture, special; on [engineering,software]]} \end{aligned}$$

Prepositions (e.g., *on* in the last example) may optionally be kept for further semantic analysis, although their use is usually dropped for simplicity. It must be noted, however, that *the spaceman on the ship* enjoys a different view than *the spaceman outside the ship* and *the spaceman without ship* is probably not even in space. The impact of prepositions on retrieval performance is not well established, but their careful treatment may be beneficial. Their use and meaning can always be postponed until the matching of PFs. Prepositions, conjuctions and other such lexical items were considered as connectors in the characterization language of *index expressions* [29].

The noun phrase presents only a few opportunities for syntactical normalization. For the verb phrase, more normalizations can be found that preserve its meaning (or rather do not lose information obviously relevant for retrieval purposes). To begin with the kernel, the elimination of time, modality, and voice seems resonable. The obviously meaningful head–modifier combinations are $[subj, verb]$ and $[verb, comp]$.

Definition 6 (verb phrase normalization I)

$$VP = subj\ kernel\ comp^* \mapsto [subj, verb(kernel)]; [verb(kernel), comp^*]$$

where the function verb(kernel) returns the main verb of the kernel.

For example

$$\textit{the students will probably attend a special lecture on Monday} \mapsto$$
$$\mapsto \texttt{[the students, attend] ; [attend, a special lecture; on Monday]}$$

In definition 6 the adverbs of the kernel are eliminated. Small experiments have suggested that adverbs have a little indexing value [28]. They might be more useful, however, if they combine with the verbs (or adjectives in the case of noun phrase) they modify; for example, `[attend, probably]`. The indexing value of such verb–adverb and adjective–adverb pairs has to be evaluated empirically.

The possibility exists to map verbs to nouns (*nominalization*) or vice versa (*verbalization*). Such normalization allows the matching of PFs derived from different sources (verb phrases or noun phrases). For example, *(to) implement* can be nominalized to *implementation*. Since the opposite transformation is also possible for nominalized verb forms, the choice has to be made on the basis of experimentation. We will presently choose to turn everything into "pictures" (noun phrases) by applying the former alternative. This results in a more drastic (and compact) normalization:

Definition 7 (Verb Phrase Normalization II)

$$VP = subj\ kernel\ comp^* \mapsto [nom(verb(kernel)), subj\ comp^*]$$

where the function nom(verb(kernel)) nominalizes the main verb of the kernel.

For example

$$\textit{the students will probably attend a special lecture on Monday} \mapsto$$
$$\mapsto \texttt{[attendance, the students; a special lecture; on Monday]}$$

Similarily, adverbs may be mapped onto adjectives to modify the nominalized verbs; for example, `[attendance, probable]`. Cross part-of-speech transformations such as those controlled by syntax can deal to some extent with derivational morphology, compensating for the conservative nature of lemmatization described in the previous section. The further application of the noun phrase normalization to the last phrase frame eventually results in

```
[attendance, student; [lecture, special]; on [Monday]]
```

All these normalizations are rather language-dependent, and the final decision of what has to be included in the phrase frames should be left to the linguists and system designers; we have merely suggested some obvious ones.

5.3 Lexicosemantical Normalization

This kind of normalization depends on the observation that certain relations can be found between the meaning of individual words. The most well known of those lexicosemantical relations are

- *synonymy* and *antonymy*
- *hyponymy* and *hypernymy* (the *is-a* relation)
- *meronymy* and *holonymy* (the *part-of* relation)

Two important aspects that should be taken into account for this kind of normalization are *polysemy* and *collocations*.

A word is polysemous if its meaning depends on the context. For example, by itself the noun *note* can be meant as a being a short letter, or as a musical note; consequently its context has to clarify its meaning. The intended meaning determines the words that are lexicosemantically related to the initial word. Using the synonymy relation for the first meaning we can obtain *brief*, while *tune* is obtained in the second case. This suggests that the conceptual context of a word should be taken into account.

Collocations are two or more words that often co-occur adjacent to one another (e.g., *health care*) having a certain meaning. When using WordNet in expanding a query with hypernyms, the notion *health care* obtains *social insurance*, which cannot be obtained in any case by expanding the two separate words. This observation suggests that collocations should be considered as single units.

Assuming that the word sense ambiguity originating from polysemy is resolved, three possibilities can been seen for lexicosemantical normalization.

1. **Semantical clustering** in analogy with stemming. For instance, several synonyms in a context are reduced to one *word cluster*. The word cluster may be represented by the most frequent of the synonyms.
2. **Semantical expansion**, expanding a term with all its -nyms. The derived terms may be weighted according to their relation with the initial term.
3. **Incorporation of a semantical similarity function into the retrieval function (fuzzy matching)**. Based on a semantical *taxonomy*, an *ontology*, or a *semantical network* we can define a *semantical similarity function* for words.

Semantical clustering is rather aggressive and suffers from the same drawbacks as stemming. For example, two "synonyms" are always overlapping in meaning and they do not actually mean the same thing. The convention to call them synonyms depends on the degree of overlap. One of the questions is how extented these clusters should be; that is, what maximum semantical distance is allowed for two words in order for them to belong to the same cluster. Again, usability and discrimination come to play an important role here. Clusters that are too large will be assigned as indexing terms to too many documents and therefore are not discriminating. Clusters that are too small (e.g., one or two words) will not have a great impact in performance compared to conventional indexing; thus they are not usable. Experimentation should provide a usably discriminating cluster size. Semantical expansion can partly overcome the cluster size problem by supplying many related terms weighted inversely proportional to their semantical distance from the original term. Expansion can easily result in an explosion of indexing or query terms, however. The possibility of fuzzy matching seems elegant and exciting, although it is far more computationally expensive than the others.

Working out fuzzy matching a bit more, using only the relations *SYN*onymy, *HYPON*nymy, and *HYPER*nymy between two words x and y, one could define

$$sim(x,y) = \begin{cases} 1 & x = y \\ 0.9 & x \in SYN(y) \\ 0.7^n & x \in HYPON_n(y) \\ 0.5^n & x \in HYPER_n(y) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $a \in HYPER_n(b)$ means that a can be found by walking in the graph of hypernyms of b a number of n steps; $a \in HYPON_n(b)$ is similarly defined. SYN is a symmetric relation, meaning that if $x \in SYN(y)$ then $y \in SYN(x)$, so it is sufficient to check only if one of the two holds. It should be noted that sim assumes an order in its arguments, x is a word from a document, and y is from a query. Moreover, hypernyms of query terms are matched with lower weights than hyponyms to reflect the assumption that a user's query *salmon* should not retrieve many documents about *fish* in general, but *fish* should retrieve documents about *salmon*.

As an example of fuzzy matching, consider the sentence

The students will probably attend a conference on software engineering.

from which, after syntactical and morphological normalization and the elimination of some (assumed) redundant elements, the following phrase frame may be constructed:

`[attendance, student; conference; [engineering, software]]`

Now let us consider another sentence

The pupils are listening carefully to the tutorial about software engineering.

which in a phrase frame representation becomes

`[listening, pupil; tutorial; [engineering, software]]`

Note that `listening` here represents the nominalized form (*the listening*) of the verb *to listen* rather than its progressive form. Using WordNet's lexical graph, and assuming that the latter sentence is part of a natural language description of a user's information need (query), the following relations hold

$$\texttt{student} = SYN(\texttt{pupil}) \Rightarrow sim(\texttt{student}, \texttt{pupil}) = 0.9$$

$$\texttt{conference} = HYPER_2(\texttt{tutorial}) \Rightarrow sim(\texttt{tutorial}, \texttt{conference}) = 0.5^2$$

The nouns *listening* and *attendance* may be matched through the relation that holds between their corresponding verbs.

$$\texttt{attend} = HYPON_1(\texttt{listen}) \Rightarrow sim(\texttt{attend}, \texttt{listen}) = 0.7$$

Using these relations, it is now easy to match the two sentences. However, this example is conveniently selected as it results in phrase frames with similar structures. In general, this is not the case, suggesting that such a lexicosemantical similarity function should be a part of a larger structural matching technique.

6 Weighting and Matching

Term weighting is a crucial part of any IR system. Statistical weighting schemes such as *tf.idf*, which perform well for single terms, do not seem to extend on multiword terms. Most work on the use of multiword indexing terms in IR concentrated on representation and matching strategies. Little consideration was given to phrase weighting and to scoring of documents matched. An obvious weighting strategy for phrasal terms is to weight a term as a function of the weights of its components. However, such strategies did not produce uniform results [18, 30]. We suggest a simple weighting scheme suitable for phrase frames that takes into account the modification structure and its depth.

Phrase frames may contain nested phrase frames (subframes) at different depths. To simplify the structural matching of complicated phrase frames, the strategy of *unnesting* can be followed. The unnesting of a phrase frame produces all possible subframes down to single-term frames. This can be understood more easily by visualizing a phrase frame as a tree; the root node is the main head, and every node is modified by its child nodes. Such an abstract tree is depicted in figure 3. Unnesting produces all possible triangles q of all possible

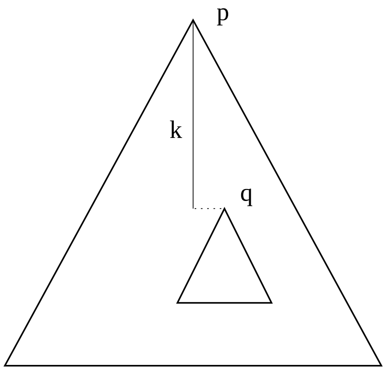

Figure 3: Tree visualization of a phrase frame p with a subframe q at depth k.

sizes and depths. The main head of a frame carries the most semantic information of all the other elements in the frame. The other elements modify the head, increasing the amount of semantic information carried by the frame. The amount of information added to the frame by an element is inversely proportional to the depth of the element within the frame.

First we introduce the predicate $sub(p, q, k)$ as a shorthand for the expression: phrase frame p has phrase frame q as a subframe at depth k. The *depth weight* of subframe q obtained from frame p can be expressed as

$$dw(q,p) = \sum_{k:sub(p,q,k)} \frac{1}{1+k}$$

The sum accounts for more than one occurrence of subframe q within p (rather rare because stylistic considerations for natural language do not favor repetitions of the same subphrase within a NP or VP). Let document d have the set $C(d)$ of phrase frames as characterization, augmented with all the unnested terms down to single terms. Then the *frame frequency* of q within document d can be described as

$$ff(q,d) = \sum_{p \in C(d)} dw(q,p)$$

The *geometrical length* of the document frame vector in the N-dimensional frame space is

$$l(d) = \sqrt{\sum_{q \in C(d)} \mathit{ff}(q,d)^2}$$

The *weight* of frame q within document d is estimated by

$$w(q,d) = \mathit{ff}(q,d)/l(d)$$

The similarity between document d and query q then can be estimated by the dot product formula

$$S(d,q) = \sum_{r \in C(d) \cap C(q)} w(r,d) * w(r,q)$$

Using the last formula, the documents of a collection can be ranked in a response to a query.

7 A Linguistically Motivated Retrieval System Architecture

In this section we discuss how a linguistically motivated retrieval system like the one described in this article can be implemented. Until now we assumed that an immaculate linguistic analysis is available, disregarding technical implementation details. However, trying to put such a retrieval system into practice, the inefficiencies and ineffectiveness of currently available NLP techniques become apparent. A major source of ineffectiveness is *linguistic ambiguity*, some of which can be resolved, while the rest requires sophisticated semantic analysis. Furthermore, NLP can be so time-consuming that it becomes impractical for real-world applications. Lacking deep semantic analysis, some design decisions have to be made in order to make a linguistically motivated retrieval system usable in the real world.

Given a collection of text documents, the indexing task assigns to each document a characterization in the form of (weighted) phrase frames. Phrase frames are derived from documents through a sequence of processing steps.

1. Tokenization.
2. Part-of-speech tagging.
3. Morphological normalization.
4. Collocation identification.
5. Lexicosemantical normalization.
6. Syntactic analysis.
7. Syntactical normalization.
8. Weighting.

The tokenization step constitutes the detection of sentence boundaries followed by the division of sentences into words. This may sufficiently be implemented based on capitalization rules, spacing, tabbing, and document layout considerations.

Part-of-speech tagging assigns a part-of-speech label to each word in a text, depending on the labels assigned to the words around it. It is possible that more than one label can be assigned to a word, suggesting some kind of *lexical ambiguity* in the input. A simple way to overcome this ambiguity is to retain only the most probable label for an ambiguous word, based on the occurrence frequencies of the word under all its possible parts of speech. Another solution would be to postpone lexical ambiguity resolution until syntactic analysis. Syntactic rules are able to resolve some lexical ambiguity, but not all. Taking collocations as single units may also resolve some lexical ambiguity. For example, while *social* can be either adjective or noun, *social security* taken as a single unit is a noun collocation because it functions as a noun. After part-of-speech tagging, morphological normalization is performed, guided by the assigned labels.

Static collocation lists or word co-occurrence statistics can be used to identify collocations. Identified collocations are treated as single units in subsequent processing steps. Lexico-semantical normalization is the following step, assuming that it is implemented by semantical clustering or expansion. If it is implemented as a semantical similarity function, then it is performed during the matching of documents to queries rather than during indexing.

Syntactic analysis or parsing reveals syntactic relations between words, collocations, and phrases in a sentence. Syntactic relations are identified based on syntactic rules (grammar). Given the part-of-speech information for a text, syntactic rules can be formulated for sequences of part-of-speech labels; for example, the combination adjective–noun surrounded by other part-of-speech labels is a noun phrase. Structural ambiguity — what modifies what — may occur during analysis. For instance, every noun phrase with three or more words, two or more of which are nouns, is a potential source of structural ambiguity. To disambiguate such structures, statistical methods can be applied. In the case of noun phrases, first, frequency information is collected from the corpus for all two-word noun phrases. Then all three-word noun phrases are disambiguated by assigning to them the most probable structure based on the frequencies of two-word noun phrases. Gradually this can be applied up to n-word noun phrases based on the frequencies of all previously disambiguated k-word noun phrases ($k < n$). *Left-dependence* may be assigned where not enough frequency information is available, since it is the most probable modification structure in the English noun phrase. A similar statistical approach can be developed to resolve the *prepositional phrase attachment* problem, guided by subcategorization information about nouns and verbs.

The next step, syntactical normalization, may be incorporated in the parser in a way that the parser outputs regularized parse tree representations (e.g., phrase frames). As soon as the collection of documents is translated to a phrase frame representation, phrase frames can be weighted according to their frequency characteristics and structure.

A similar procedure to the above indexing steps can be followed to turn a natural language query into a phrase frame representation, allowing the matching of queries to documents. The indexing procedure just described can replace the indexing part of a conventional retrieval system architecture. There is no obvious need why radical architectural changes should be made. Inverted files, vector space, and probabilistic retrieval models are still suitable and may be adapted to work with linguistically-motivated indexing terms. What really changes is the way that indexing terms are extracted from documents and how these are matched. The current inefficiencies and ineffectiveness of NLP techniques can be treated for the time being by such statistical solutions as the (crude) ones described above. Fortunately, the explosion in computational power that becomes available daily, combined with the efforts put into NLP issues from the (computational) linguists' side, suggests that the use of linguistically motivated retrieval systems in everyday practice is merely a matter of time.

8 Conclusions

The bag-of-words paradigm has dominated commercially available information retrieval systems for about three decades. The main reasons for the endurance of such systems based on such simple assumptions as the naive keyword retrieval hypothesis are first, that they are relatively easy and simple to implement (it takes a third-year computer science student with the knowledge of a programming language, an IR textbook, and some days' time), and second and most important, that these systems have presented until recently a satisfactory effectiveness in searching collections in the class of megabytes.

The digital and networking revolution has made available data in the class of gigabytes, exposing the inadequate nature of keyword-based systems. The searching for information has become a laborious task for a user who presently has to perform her or his own selection over the "dirty" output of a World Wide Web search engine, for example. As a consequence, many researchers have aimed at higher levels of natural language utilization in IR, assuming that better "understanding" the information need as well as the information residing in a database is the key for improving retrieval effectiveness.

The attempts made to break out of the bag-of-words paradigm by employing NLP and other linguistic resources have until now presented inconsistent or at least dubious results, however. One explanation of why NLP has not had more successes in IR is that it does not go far enough. First, the currently available NLP techniques suffer from lack of accuracy and efficiency, and second, there are doubts if syntactic structure is a good substitute for semantic content. The evidence so far suggests further investigation and better modeling.

In this article, we have reviewed some of the most important research in the field, and discussed a general model for a linguistically motivated retrieval system. We believe that a retrieval schema based on the phrase retrieval hypothesis and incorporating linguistic normalization has more potential in improving retrieval effectiveness than keyword-based schemas. We have suggested a suitable model and some techniques, however, whether or not the discussed techniques work remains unclear and the answer requires more empirical data.

Considering that better IR means more user satisfaction, perhaps a more radical change in the focus of IR is needed. Maybe the future of IR is not to provide better ranking of retrieved documents but to supply the very information a user is seeking. A compact summary of retrieval results or a brief answer might be more usable for an average user than a ranked list of hundreds of documents. Automatic summarization, question answering, and information extraction systems require advanced NLP techniques, however. Furthermore, the traditional precision- and recall-based retrieval quality metrics may not be able to evaluate the ability of a system to derive such information; consequently other metrics will have to be developed. Nevertheless, one thing seems certain for the future: NLP and other linguistic resources will become — if they are not already becoming — indispensable parts of every effective IR system.

References

[1] A. T. Arampatzis, T. Tsoris, C. H. A. Koster, and Th. P. van der Weide. Phrase-based Information Retrieval. *Information Processing & Management*, 34(6):693–707, December 1998.

[2] J. B. Lovins. Development of a Stemming Algorithm. *Mechanical Translation and Computational Linguistics*, 11(1):22–31, 1968.

[3] M. F. Porter. An Algorithm for Suffix Stripping. *Program*, 14(3):130–137, 1980.

[4] Donna Harman. How Effective is Suffixing? *Journal of the American Society for Information Science*, 42(1):7–15, 1991.

[5] David Hull. Stemming Algorithms — A Case Study for Detailed Evaluation. *Journal of the American Society for Information Science*, 47(1), 1996.

[6] Robert Krovetz. Viewing Morphology as an Inference Process. In R. Korfhage, E. Rasmussen, and P. Willett, editors, *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 191–202, Pittsburgh, PA, USA, 1993. ACM Press.

[7] M. Popovic and P. Willett. The Effectiveness of Stemming for Natural Language Access to Slovene Textual Data. *Journal of the American Society for Information Science*, 43(5):384–390, 1992.

[8] Wessel Kraaij and Renée Pohlmann. Viewing Stemming as Recall Enhancement. In Hans-Peter Frei, D. Harman, P. Schauble, and Ross Wilkinson, editors, *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 40–48, Zurich, Switzerland, 1996. ACM Press.

[9] G. A. Miller. WordNet: A Lexical Database for English. *Communications of the ACM*, 38(11):39–41, 1995.

[10] A. F. Smeaton, F. Kelledy, and R. O'Donnell. TREC-4 Experiments at Dublin City University: Thresholding Posting Lists, Query Expansion with WordNet and POS Tagging of Spanish. In Donna K. Harman, editor, *The Fourth Text REtrieval Conference (TREC-4)*, pages 373–390, Gaithersburg, Maryland, 1995. Department of Commerce, National Institute of Standards and Technology (NIST) Special Publication 500-236.

[11] Ellen M. Voorhees. Query Expansion using Lexical-Semantic Relations. In W. Bruce Croft and C. J. van Rijsbergen, editors, *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 61–69, Dublin, Ireland, June 1994. ACM Press.

[12] Wessel Kraaij and Renée Pohlmann. Using Linguistic Knowledge in Information Retrieval. Technical Report OTS Working Paper OTS-WP-CL-96-001, OTS, Utrecht, The Netherlands, 1996.

[13] Allan Smeaton and Ian Quigley. Experiments on using semantic distances between words in image caption retrieval. In H. Frei, Donna Harman, P. Schäuble, and R. Wilkinson, editors, *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 174–180. ACM Press, 1996.

[14] R. Richardson and A. F. Smeaton. Using wordnet in a knowledge-based approach to information retrieval. In *Proceedings of the 17th BCS-IRSG Colloquium on Information Retrieval*, Crewe, 1995.

[15] Julio Gonzalo, Felisa Verdejo, Irina Chugur, and Juan Cigarrán. Indexing with wordnet synsets can improve text retrieval. In *Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, Montreal, 1998.

[16] M. Sanderson. Word Sense Disambiguation and Information Retrieval. In W. Bruce Croft and C. J. van Rijsbergen, editors, *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 142–151, Dublin, Ireland, June 1994. ACM Press.

[17] Donna K. Harman, editor. *The Sixth Text REtrieval Conference (TREC-6)*. Department of Commerce, National Institute of Standards and Technology (NIST) Special Publication 500-240, Gaithersburg, Maryland, 1997.

[18] Joel L. Fagan. *Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-Syntactic Methods*. PhD thesis, Department of Computer Science, Cornell University, Ithaca, New York 14853-7501, 1987.

[19] Mandar Mitra, Chris Buckley, Amit Singhal, and Claire Cardie. An Analysis of Statistical and Syntactic Phrases. In *Proceedings of RIAO'97 Computer-Assisted Information Searching on Internet*, pages 200–214, McGill University, Montreal, Canada, 1997.

[20] Wessel Kraaij and Renée Pohlmann. Comparing the Effect of Syntactic vs. Statistical Phrase Indexing Strategies for Dutch. In Christos Nicolaou and Constantine Stephanidis, editors, *Proceedings of Second European Conference on Research and Advanced Technology for Digital Libraries ECDL'98*, pages 605–614, Heraklion, Crete, 1998. Springer-Verlag.

[21] T. Strzalkowski and J. P. Carballo. Natural Language Information Retrieval: TREC-4 Report. In Donna K. Harman, editor, *The Fourth Text REtrieval Conference (TREC-4)*, pages 245–258, Gaithersburg, Maryland, 1995. Department of Commerce, National Institute of Standards and Technology (NIST) Special Publication 500-236.

[22] C. Zhai, X. Tong, N. M. Frayling, and D. A. Evans. Evaluation of Syntactic Phrase Indexing — CLARIT NLP Track Report. In Donna K. Harman and Ellen M. Voorhees, editors, *The Fifth Text REtrieval Conference (TREC-5)*, pages 347–358, Gaithersburg, Md. 20899, 1996. Department of Commerce, National Institute of Standards and Technology (NIST) Special Publication 500-238.

[23] A. F. Smeaton, R. O'Donnell, and F. Kelledy. Indexing Structures Derived from Syntax in TREC-3: System Description. In Donna K. Harman, editor, *The Third Text REtrieval Conference (TREC-3)*, pages 55–67, Gaithersburg, Md. 20899, 1994. Department

of Commerce, National Institute of Standards and Technology (NIST) Special Publication 500-225.

[24] W. Winograd. *Language as a Cognitive Process.* Addison-Wesley Pub. Co., Reading MA, USA, 1983.

[25] A. T. Arampatzis, T. Tsoris, and C. H. A. Koster. IRENA: Information Retrieval Engine based on Natural language Analysis. In *Proceedings of RIAO'97 Computer-Assisted Information Searching on Internet*, pages 159–175, McGill University, Montreal, Canada, 1997.

[26] D. A. Hull, G. Grefenstette, B. M. Schulze, E. Gaussier, H. Schutze, and J. O. Pedersen. Xerox TREC-5 Site Report: Routing, Filtering, NLP, and Spanish Tracks. In D. K. Harman and E. M. Voorhees, editors, *The Fifth Text REtrieval Conference (TREC-5)*, pages 167–180, Gaithersburg, Md. 20899, 1996. Department of Commerce, National Institute of Standards and Technology (NIST) Special Publication 500-238.

[27] Tomek Strzalkowski, Fang Lin, Jin Wang, and Jose Perez-Carballo. Evaluating natural language processing techniques in information retrieval — a trec perspective. In Tomek Strzalkowski, editor, *Natural Language Information Retrieval.* Kluwer Academic Publishers, Dordrecht, April 1999.

[28] A. T. Arampatzis, Th. P. van der Weide, C. H. A. Koster, and P. van Bommel. Text Filtering using Linguistically-motivated Indexing Terms. Technical Report CSI-R9901, Computing Science Institute, University of Nijmegen, Nijmegen, The Netherlands, January 1999.

[29] P.D. Bruza and Th.P. van der Weide. Stratified Hypermedia Structures for Information Disclosure. *The Computer Journal*, 35(3):208–220, 1992.

[30] David D. Lewis and W. Bruce Croft. Term clustering of syntactic phrases. In *Proceedings of the 13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 385–405. ACM Press, 1990.

Readings for Further Study

[1] G. Salton and M.J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill, New York, New York, 1983.

[2] C. J. van Rijsbergen. *Information Retrieval.* Butterworths, London, United Kingdom, 1990.

[3] Karen Sparck Jones and Peter Willett, editors. *Readings in Information Retrieval.* Academic Press/Morgan Kaufmann, 1997.

[4] Tomek Strzalkowski, editor. *Natural Language Information Retrieval.* Kluwer Academic Publishers, Dordrecht, April 1999.