CS 92-05

SEXTANT:
Extracting Semantics
from Raw Text
Implementation Details

Gregory Grefenstette
Computer Science Department
University of Pittsburgh
Pittsburgh, PA 15260
gffen@cs.pitt.edu

February 28, 1992

Abstract

Until the recent past, there have been two extreme approaches toward extracting knowledge from text. On one hand Artificial Intelligence systems have long operated under the assumption that an a priori structuring of the text domain is necessary before treating the text. On the other, the Information retrieval community has limited itself to simple document co-occurrence statistics as sole indicator of word semantics. The first approach is difficult to extend, while the second relies on context which is too coarse-grained. We present here SEXTANT, a complete system that uses finer-grained syntactic contexts to discover similarities between words. The system is based on the hypothesis that words that are used in a similar way throughout a corpus are indeed semantically similar. This system takes raw text in input, performs syntactic analysis to extract fine-grained contexts, compares contexts of words, and produces a list of similar words as output. No domain structuring is needed. Initial results show that these similar words can improve classical information retrieval, through use in query expansion. This article will present a detailed description of the SEXTANT system.
1 Introduction

There are two types of approaches to extracting information from text by computer. The knowledge-rich approaches such as those found in FERRET [Mau91] and TRUMP [JZ88] suppose that a rich network of hand-crafted knowledge structures exist for the textual domain, before the knowledge extraction takes place. The expense and complexity of such approaches have reduced their practical applicability [LGP+91]. Knowledge-poor approaches have been taken by researchers in Information Retrieval, from [Sal71] to [Cro90]. These approaches have almost exclusively used the statistics of document co-occurrence in order to cluster similar word or noun phrases. Since two terms appearing in the same document have some semantic relatedness, the hypothesis goes, document co-occurrence counts should provide an indication of semantic similarity among terms in a large corpus. This statistic, however, has not proven in practice to be of great use [PW91] in improving information retrieval.

We have developed a system called SEXTANT (Semantic EXtraction from Text via Analyzed Networks of Terms), which exploits previously unexplored syntactic contexts to produce term association lists which could be used for automatic thesaurus construction or query expansion. The method employed is fully automatic and can be applied to large corpora without previous hand-coding. The semantic information extracted by this method has been shown to be more useful in information retrieval than that produced by document co-occurrence techniques [Gre92]. Here we will present the implementation details of SEXTANT, and provide examples of its outputs on various corpora.

2 Overview of the SEXTANT system

The SEXTANT system takes as input raw text. The text is divided into words. Individual words are looked up in a dictionary which supplies their possible grammatical categories, which we will call tags. A choice of likely categories for each word in the corpus is made using the local context around the word. Then a number of simple algorithms pass over each sentence, dividing them into noun phrases and verb phrases, connecting modifiers to head nouns, and connecting verbs to subjects and objects. Whenever an ambiguity appears all interpretations are retained.

At this point, for each noun in the corpus, a certain number of adjectives, other nouns, and verbs have been found to modify it. Though a number of items may have been missed due to errors introduced at any stage: typographical errors, absence from the dictionary, improper grammatical disambiguation, incorrect attachment to other words, over a large corpus many correct modifiers will have been found.

Now, borrowing from social sciences, similarity measures are applied to nouns using the words that modify them as their attributes. Words which
share a great number of automatically weighted attributes are found as being similar. Lists of similar words are the output that SEXTANT produces.

What has been done by SEXTANT then is what humans do, given an unknown word, the context of the word, i.e., what words have been found to modify it, give a clue to its meaning. Humans make use of much richer contexts, involving deeper semantic models of the modifiers, whereas SEXTANT can only find when two words are used in a similar manner over a large corpus. But SEXTANT using only surface syntactic similarity often finds words sharing obvious semantic relations.

3 SEXTANT Details

We will now present in detail all the steps that SEXTANT performs on an input text to produce its output, illustrated by examples.

3.1 Morphological Analysis

Raw input text is divided into words using a regular grammar (Unix’s lex) which separates words using spaces and punctuation as delimiters. A certain number of common abbreviations such as ‘d, ‘m, ‘ll, ‘re, ‘ve as well as the ‘s are broken apart without expansion from the word preceding them. A period is considered as a separator when it is not in a sequence such as letter-period-letter-period-letter..., or in a number.

Words are looked up in a dictionary for their parts of speech. In our system, we use the morphological package described above and a dictionary, both developed for the CLARIT project [EHL91], which assigns a limited number of syntactic categories such as SINGULAR- NOUN, PLURAL-NOUN, ADJECTIVE, GERUND, AUXILIARY-HAVE, PLURAL-VERB-ACTIVE, PAST-PARTICIPLE, etc. to each word found in the dictionary. Words not found in the dictionary are assigned a default category of NOUN.

3.2 Syntactic Disambiguation

After dictionary lookup, a word may possess more than one category. This is normally the point where a syntactic analysis takes over, producing a parse tree. In SEXTANT, this is not the case. Instead, we use a disambiguator developed by D. C. Leberknight at the Laboratory for Computational Linguistics at Carnegie-Mellon, that implements a time linear stochastic grammar based on Brown corpus frequencies. This disambiguator uses the frequencies that a category appears with a given word in the Brown corpus, as well as the frequency of one category following another in the Brown corpus to produce most probable sequences of syntactic categories through a given sentence.
3.3 Noun and Verb Phrase Extraction

Once each word has been disambiguated to a single grammatical category, or tag, each sentence is broken up into noun phrases and verb phrases using another linear time algorithm. SEXTANT extracts complex noun phrases including prepositions and conjunctions to produce the longest possible noun phrase.

Noun phrase isolation is performed in the following manner. A matrix, called CanContinue, is manually created with rows and columns representing all the possible grammatical categories provided by the dictionary. Each entry is either NO or YES, indicating if the sequence of categories represented by that row followed by that column can be part of a noun phrase. For example the entry corresponding to CanContinue(DETERMINER, NOUN) has a YES in it, and the entry corresponding to CanContinue(DETERMINER, PLURAL-VERB-ACTIVE) has a NO. Two other vectors are used, CanBegin and CanEnd. Each element of CanBegin is YES if the corresponding grammatical category can begin a noun phrase, and NO if not. CanEnd encodes similar information for ending a phrase. For example, CanBegin(DETERMINER) is YES since it can begin a noun phrase, and CanEnd(DETERMINER) is NO since a noun phrase can not end with a determiner.

The algorithm for extracting phrases is then:

\[
\text{currentPhrase} = 1 \\
\text{InPhrase} = \text{NO} \\
i = 1 \\
\text{while } (i \leq \text{NumberOfWordsInSentence}) \text{ do} \\
\quad \text{if } (\text{InPhrase} == \text{YES}) \text{ then} \\
\quad\quad \text{if } \text{CanContinue(Tag[i-1], Tag[i])} \text{ then} \\
\quad\quad\quad \text{Phrase}[i] = \text{currentPhrase} \\
\quad\quad\quad \text{next } i \\
\quad\quad \text{else} \\
\quad\quad\quad \text{find last word } j \text{ in current phrase} \\
\quad\quad\quad \text{for which } \text{CanEnd}(j) == \text{YES} \\
\quad\quad\quad \text{currentPhrase}++ \\
\quad\quad\quad i = j + 1 \\
\quad\quad \text{InPhrase} = \text{NO} \\
\quad\text{fi} \\
\quad\text{else} \\
\quad\quad \text{if } \text{CanBegin(Tag[i])} \text{ then} \\
\quad\quad\quad \text{Phrase}[i] = \text{currentPhrase} \\
\quad\quad\quad \text{InPhrase} = \text{YES} \\
\quad\quad\text{fi} \\
\quad \text{next } i \\
\text{fi} \]

4
endwhile

A similar set of matrix and vectors exists for isolating verb phrases.

Example

As a running example, we will consider treatment of the following sample text from a medical corpus:

It was concluded that the carcinoembryonic antigens represent cellular constituents which are repressed during the course of differentiation of the normal digestive system epithelium and reappear in the corresponding malignant cells by a process of derepressive dedifferentiation.

The first two stages of morphological analysis and grammatical disambiguation are performed by the CLARIT system [EHL91] as described in sections 3.1 and 3.2. Once the noun and verb phrases have been isolated (marked NP and VP below), the sample now becomes:

NP it
VP be conclude
-- that
NP the carcinoembryonic antigen
VP represent
NP cellular constituent
-- which
VP be repress
NP during the course of differentiation of the
normal digestive system epithelium
-- and
VP reappear
NP in the correspond
NP malignant cell by a process of derepressive
differentiation

3.4 Fine-Grained Syntactic Relations

Once the original sentence has been divided into noun phrases and verb phrases, syntactic relations between words within these phrases, and across these phrases are extracted by a five pass method that will be described here.

3.4.1 Pass One: Noun Phrases Left-to-Right

First, noun phrases are scanned from left to right attaching modifiers such as articles, adjectives and adjectively used nouns to the farthest noun appearing before a preposition.

The algorithm for performing this attachment, originally implemented in [Gre83], is very simple. Three sets of tags are used: a StartSet of tags which
can modify another word, a receiveSet of tags which can be modified by a
member of the StartSet, and a set of tags recognized as Prepositions.

\[
i = \text{FirstWordInNounPhrase}
\]
while (i <= LastWordInNounPhrase)
  if ( Tag[i] in StartSet) then
    j = i+1
    while (j <= LastWordInNounPhrase and Tag[j] not Preposition)
      if (Tag[j] in ReceivingSet) then
        CreateRelation between words i and j
      fi
    next j
  fi
next i
endwhile

Any ambiguities are retained, so this algorithm tends to create more rela-
tions than would be produced by a human, or by a more intelligent system,
for example in the phrase 'a red table top' there will be a relation created
between red and table as well as between red and top and between table and
top, but over a large corpus, one of either red table or red top will probably
appear more often than the other.

The relations created depend upon the tags attached to the words being
related. In SEXTANT, the following useful relations are recognized:

- ADJ: an adjective modifies a noun (e.g., a large table)
- NN: a noun modifies a noun (e.g., a table top)
- NNPREP: a noun that is the object of a proposition modifies a preceeding
  noun (e.g., the top of the table)
- SUBJ: a noun is the subject of a verb (e.g., the table shook)
- DOBJ: a noun is the direct object of a verb (e.g., the table was shaken)
- IOBJ: a noun is the indirect object of a verb. (e.g., The book was placed
  on the table)

Of these relations, ADJ and NN are recognized during this first pass. Informa-
tion as to whether a noun is modified by an article or by a preposition is also
recorded during this phase.

Example

During this pass over the previously given text, the following relations are ex-
tracted:
antigen, carcinoembryonic < ADJ
constituent, cellular < ADJ
digestive, normal < ADJ
system, normal < ADJ
epithelium, normal < ADJ
system, digestive < NN
epithelium, digestive < NN
epithelium, system < NN
cell, malignant < ADJ
dedifferentiation, derepressive < NN

3.4.2 Pass Two: Noun Phrases Right-to-Left

After the first pass, the head noun of any noun phrase or prepositional phrase remains unattached. There may be articles, or modifiers or prepositions attached to it, but it remains free to be attached to something else. The purpose of this right-to-left pass is to attach the head nouns of prepositional phrases to a free noun appearing before it.

The problem of prepositional phrase attachment can be complex, as can be seen in the well-known sentence: He saw the girl on the hill with the telescope in which with the telescope may be modifying hill, or girl, or saw. Here we simplify the information that we can accurately extract by retaining only the closest relation with another noun, so that from that sentence, SEXTANT retains only the relation between telescope and hill.

The algorithm for this phase is:

i = LastWordInNounPhrase
while (i > FirstWordInNounPhrase)
    if (word i is unattached) then
        if (word i is modified by a preposition)
            j = word before this preposition
        else
            j = i - 1
        fi
        Find the first noun from j downto FirstWordInNounPhrase
        and create a relation between i and this noun
    fi
    i = i - 1
endwhile

While this algorithm proceeds to find a previous noun to attach an unattached noun to, record is kept of any prepositions found along the way. If a preposition is discovered before the noun, then a NNPREP relation is created, otherwise a NN relation, for a noun in apposition, is created.
Example

During this pass over the previously given text, the following relations are extracted:

course, differentiation < NNPREP
differentiation, epithelium < NNPREP
cell, process < NNPREP
process, dedifferentiation < NNPREP

3.4.3 Pass Three: Verb Phrases Right-to-Left

After the first two passes, there are usually unattached nouns before and after each verb phrase. The next two passes will attempt to attach verbs to these subjects and objects.

Before Pass Three, each verb phrase is analyzed to find the head verb and to determine if the phrase is active or passive. This analysis is simple: Trace the verb phrase to its last verb, this becomes the head verb. A verb phrase begins as ACTIVE. If an auxiliary verb form of BE is found the verb phrase is PASSIVE. If an -ING verb is found (other than being), then the phrase becomes ACTIVE. If the head verb is a form of BE, then the verb phrase becomes ATTRIBUTIVE.

Once the verb phrase has been analyzed, the parser of SEKTANT searches for all free nouns before the phrase to become the subjects (or direct objects if the verb phrase is PASSIVE). The search stops when another verb phrase is encountered.

3.4.4 Pass Four: Verb Phrases Left-to-Right

During Pass Four, a similar search takes place a verb phrase to find an unattached noun which becomes the direct object for a ACTIVE verb phrase. The first head noun of a prepositional phrase after the verb phrase becomes the indirect object of the verb.

Example

During these two passes over the sample text, the following relations are extracted:

repress, antigen < DOBJ
antigen, represent < SUBJ

represent, constituent < DOBJ
reappear, cell < DOBJ
3.4.5 Pass Five: Gerunds

A fifth pass goes through the text trying to attach gerundive adjectives to potential subjects and objects. This follows the same algorithms as passes three and four, but with the constraint relaxed that the subjects and objects found be unattached.

Example

During this pass over the previously given text, the following relations are extracted:

cell , correspond < SUBJ
correspond , cell < DOBJ

The original sample:

It was concluded that the carcinoembryonic antigens represent cellular constituents which are repressed during the course of differentiation of the normal digestive system epithelium and reappear in the corresponding malignant cells by a process of derepressive dedifferentiation.

will yield all the above relations which are fed into the next stage of similarity comparison in the following form:

antigen carcinoembryonic
antigen repress-DOBJ
antigen represent-SUBJ
constituent cellular
constituent represent-DOBJ
course repress-IOBJ
course differentiation
digestive normal
system normal
epithelium normal
system digestive
epithelium digestive
epithelium system
differentiation epithelium
corresponding reappear-IOBJ
cell correspond-SUBJ
cell malignant
cell reappear-DOBJ
cell correspond-DOBJ
cell process
dedifferentiation derepressive

3.4.6 Discussion

As can be expected by the simplicity of the algorithms for disambiguating the text, and for extracting and analyzing phrases, errors may appear among
the otherwise acceptable list of relations extracted. For example, cell is not the direct object of \texttt{reappear} and the true subject of \texttt{reappear} should be \texttt{constituents}, which is absent. These algorithms are used for reaping information through great quantities of text, and not for providing a model of human competence. They have the advantage of being very fast, since no backtracking, recursion, or maintenance of possible parses is involved. Though a great many serious linguistic problems are not addressed, such as anaphora resolution, multi-word verbs, garden paths, etc. [Smu91], it does provide roughly correct results. As we shall see, in section 3.5.1 even with these imperfections, over a large enough corpus, useful domain knowledge can be generated by SEXTANT.

3.5 Similarity Calculations

Once the syntactic analysis of the corpus is performed, each word in the corpus possesses a certain quantity of context which SEXTANT uses to judge word similarity.

Similarity between objects based upon shared attributes has been widely studied in the Social Sciences. See [Rom90] for a clear introduction. SEXTANT implements a wide variety of similarity measures described in this reference.

For the moment, work with SEXTANT has concentrated on finding similar nouns. Each noun found in the text is considered an object, and the words that are found to modify it are considered its attributes. A noun can be modified by an adjective (ADJ), by another noun (NN and NNPREP), by a verb (SUBJ, DOBJ, and IOBJ), and these modifications are taken to be the known attributes of the noun.

Similarity comparison between words is performed by the following algorithm.

\begin{verbatim}
for each noun i in the corpus
    for each noun j <> i in the corpus
        use a similarity measure to calculate
            the distance between noun i and noun j
        sort the nouns in function of the similarity to noun i
        retain the closest N words to noun i
endfor
\end{verbatim}

The similarity measure that seems to produce the best results in SEXTANT is a weighted Tanimoto [Tan58] similarity measure, also known as the Jaccard measure. The binary Tanimoto measure between two objects \(m\) and \(n\) is the number of shared attributes divided by the number of attributes in the unique union of the set of attributes for each object.

\[
\frac{\text{Count}([\text{Attributes shared by object}_m \text{ and object}_n])}{\text{Count}([\text{Unique attributes possessed by object}_m \text{ or object}_n])}
\]

10
Let's give an example. Suppose that we are comparing two nouns dog and cat possessing the following attributes derived as described in section 3.4.

dog pet-DOBJ
dog eat-SUBJ
dog shaggy
dog brown
dog leash
cat pet-DOBJ
cat pet-DOBJ
cat hairy
cat leash

Here dog has 5 attributes. Cat has 3 attributes, one of which appears twice. In a strictly binary Tanimoto measure, the similarity of cat and dog would be

\[
\frac{\text{Count}([\text{leash, petDOBJ}])}{\text{Count}([\text{brown, eatSUBJ, hairy, leash, petDOBJ, shaggy}])} = \frac{2}{6} = 0.333
\]

Moving from a binary to a weighted measure can be done in many ways. We have found useful to weight attributes using a log-entropy weighting that has been shown to improve document retrieval in [Dum91]. Each attribute is attributed a global weight between 0 and 1 in function of how many different objects it associated with, and how often it appears, using

\[
1 - \sum_j \frac{p_{ij}\log(p_{ij})}{\log(nrels)}
\]

where \( p_{ij} \) is

\[
\frac{\text{freq of attribute}_j \text{ with object}_i}{\text{nb of attributes for object}_i}
\]

and where \( nrels \) is the total number of relations extracted from the corpus. A higher global weighting means that the word appears less often in the corpus.

In order to show how the weighted Tanimoto similarity is calculated in SEXTANT, let us suppose in this example that the global weights of the attributes above, when calculated over the whole corpus, were:

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>brown</td>
<td>0.9</td>
</tr>
<tr>
<td>hairy</td>
<td>0.85</td>
</tr>
<tr>
<td>pet-DOBJ</td>
<td>0.6</td>
</tr>
<tr>
<td>cat-SUBJ</td>
<td>0.7</td>
</tr>
<tr>
<td>leash</td>
<td>0.75</td>
</tr>
<tr>
<td>shaggy</td>
<td>0.8</td>
</tr>
</tbody>
</table>

Before calculating similarity, a local weighting is also given each object-attribute pair. If an attribute appears more than once for word, as pet-DOBJ does for cat, then the weight of that attribute is multiplied by the log of its frequency for that word, in this log-entropy scheme. For example, since cat has the attribute
pet-DOBJ twice, the weight of that attribute for cat will be its global weight multiplied by its local weight, which gives \(0.6 \log(2+1) = 0.79\). The value of the attribute pet-DOBJ for dog would be \(0.6 \log(1+1) = 0.6\).

Now, although cat and dog share the attribute pet-DOBJ equally in the binary case, in the weighted case, the weight of this attribute is greater for cat. Our formula for the weighted Tanimoto similarity measure between two objects \(m\) and \(n\) is

\[
\frac{\sum_{\text{unique attributes}} \min(\text{weight}(\text{object}_m, \text{attribute}), \text{weight}(\text{object}_n, \text{attribute}))}{\sum_{\text{unique attributes}} \max(\text{weight}(\text{object}_m, \text{attribute}), \text{weight}(\text{object}_n, \text{attribute}))}
\]

Note when the weights are restricted to 0 and 1 that this last formula is equivalent to previous binary Tanimoto formula, though this is by no means the only way in which to generalize the binary formula to the weighted case.

For our cat and dog example, this weighted similarity measure gives:

\[
\frac{0 + 0 + 0 + 0.75 + 0.6 + 0}{0.9 + 0.7 + 0.85 + 0.75 + 0.79 + 0.8} = 0.28
\]

In other words, they are a little less similar than in the binary case, since cat possesses one attribute to a different degree than dog.

### 3.5.1 Output

Once the similarity of each word is calculated and the results sorted, SEXTANT produces a list of the most similar words for each word in the corpus. Since similarity is based upon recognized contexts, words appearing often in the corpus possess the best indications of their meanings.

As an example, the sample text cited above was drawn from 1 Megabytes of medical text often used as a testbed in Information Retrieval. When the whole corpus was analyzed as described in section 3.4.1 to 3.4.5, the context of the word cause was found to be the following:

- cause
- admitting
- cause
- aortic-regurgitation
- cause
- arachnoiditis
- cause
- basic
- cause
- child
- cause
- clarify-DOBJ
- cause
- clear-DOBJ
- cause
- common
- cause
- complication
- cause
- concern
- cause
- concern-IOBJ
- cause
- constriction
- ...
- cause
- ulceration
cause uncertain
cause unknown

When the similarity measure described in the previous section was applied to all the extracted nouns, cause was found to be closest to the word etiology. This is because they shared the following modifying attributes, with their global weights:

<table>
<thead>
<tr>
<th>attribute</th>
<th>Global Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>determine-DObj</td>
<td>0.53352</td>
</tr>
<tr>
<td>hydrocephalus</td>
<td>0.59494</td>
</tr>
<tr>
<td>jaundice</td>
<td>0.65286</td>
</tr>
<tr>
<td>patient</td>
<td>0.41963</td>
</tr>
<tr>
<td>possible</td>
<td>0.58791</td>
</tr>
<tr>
<td>unknown</td>
<td>0.71528</td>
</tr>
</tbody>
</table>

The results are best for those words appearing most often, since these words supply the most attributes on which comparisons can be made. The closest words found to other words in this corpus, from most frequent appearing hundreds of times are:
<table>
<thead>
<tr>
<th>word</th>
<th>closest words</th>
</tr>
</thead>
<tbody>
<tr>
<td>cell</td>
<td>tissue</td>
</tr>
<tr>
<td>patient</td>
<td>case</td>
</tr>
<tr>
<td>effect</td>
<td>response, change</td>
</tr>
<tr>
<td>study</td>
<td>change</td>
</tr>
<tr>
<td>change</td>
<td>increase</td>
</tr>
<tr>
<td>case</td>
<td>patient</td>
</tr>
<tr>
<td>level</td>
<td>concentration</td>
</tr>
<tr>
<td>result</td>
<td>effect, patient</td>
</tr>
<tr>
<td>response</td>
<td>effect</td>
</tr>
<tr>
<td>child</td>
<td>patient</td>
</tr>
<tr>
<td>group</td>
<td>patient, child</td>
</tr>
<tr>
<td>activity</td>
<td>effect</td>
</tr>
<tr>
<td>increase</td>
<td>rise</td>
</tr>
<tr>
<td>method</td>
<td>technique</td>
</tr>
<tr>
<td>treatment</td>
<td>therapy</td>
</tr>
<tr>
<td>rate</td>
<td>concentration, value, level, decrease</td>
</tr>
<tr>
<td>development</td>
<td>change, increase</td>
</tr>
<tr>
<td>disease</td>
<td>lesion</td>
</tr>
<tr>
<td>concentration</td>
<td>level, content</td>
</tr>
<tr>
<td>type</td>
<td>form</td>
</tr>
<tr>
<td>tumor</td>
<td>carcinoma</td>
</tr>
<tr>
<td>factor</td>
<td>role</td>
</tr>
<tr>
<td>lesion</td>
<td>disease</td>
</tr>
<tr>
<td>tissue</td>
<td>lung, cell, extract</td>
</tr>
<tr>
<td>test</td>
<td>technique</td>
</tr>
<tr>
<td>form</td>
<td>type, case</td>
</tr>
<tr>
<td>acid</td>
<td>concentration</td>
</tr>
<tr>
<td>period</td>
<td>time, level</td>
</tr>
<tr>
<td>reaction</td>
<td>response</td>
</tr>
<tr>
<td>value</td>
<td>level, concentration</td>
</tr>
<tr>
<td>technique</td>
<td>method</td>
</tr>
<tr>
<td>time</td>
<td>day, rate, number</td>
</tr>
<tr>
<td>therapy</td>
<td>treatment</td>
</tr>
<tr>
<td>rat</td>
<td>mouse</td>
</tr>
<tr>
<td>growth</td>
<td>effect</td>
</tr>
<tr>
<td>number</td>
<td>concentration, value, time, amount</td>
</tr>
<tr>
<td>content</td>
<td>concentration</td>
</tr>
<tr>
<td>observation</td>
<td>data, study</td>
</tr>
</tbody>
</table>
At the bottom of the list, for words appearing less than a dozen times, the similarities extracted by SEXTANT become more doubtful, as can be seen in:

<table>
<thead>
<tr>
<th>infrequent words</th>
<th>closest words</th>
</tr>
</thead>
<tbody>
<tr>
<td>gorilla</td>
<td>chimpanzee</td>
</tr>
<tr>
<td>saturation</td>
<td>svc</td>
</tr>
<tr>
<td>leukemia</td>
<td>pulmonary-emphysema, renal-failure</td>
</tr>
<tr>
<td>regeneration</td>
<td>cytomorphosis</td>
</tr>
<tr>
<td>sulfate</td>
<td>sulphate</td>
</tr>
<tr>
<td>twin</td>
<td>radiograph</td>
</tr>
<tr>
<td>al</td>
<td>et</td>
</tr>
<tr>
<td>phosphatase</td>
<td>cathepsin</td>
</tr>
<tr>
<td>bacteriophage</td>
<td>sb25</td>
</tr>
<tr>
<td>breast</td>
<td>undergoing</td>
</tr>
<tr>
<td>childhood</td>
<td>psychosis</td>
</tr>
<tr>
<td>thymidine</td>
<td>tocopherol</td>
</tr>
<tr>
<td>filtration</td>
<td>gel</td>
</tr>
<tr>
<td>reticulum</td>
<td>peer, raise, erythrom, homologous</td>
</tr>
<tr>
<td>palate</td>
<td>cleft</td>
</tr>
<tr>
<td>viii</td>
<td>concentrate</td>
</tr>
<tr>
<td>bu</td>
<td>s3</td>
</tr>
<tr>
<td>outflow</td>
<td>infundibulum</td>
</tr>
</tbody>
</table>

The interesting point about the above results is that, at least for the frequently appearing words, the words found as most similar by SEXTANT do seem to have clear semantic relations within the corpus.

That the relations extracted by SEXTANT are domain-specific can be seen by comparing similar words from two different domains. Just as related words were extracted for the medical domain above, similarities were calculated over a smaller domain devoted to Information Science abstracts. Some results of words appearing in both corpora follow:
4 Application to Information Retrieval

In order to test this intuitive feeling, an experiment was run, reported in [Gre92], in which queries from an information retrieval testbed were expanded by the words found most similar to each by SEXTANT. Then the expanded queries were used to retrieve the documents in the testbed using classic information retrieval methods. The expanded queries improved the precision and recall slightly, indicating that the expanded words were indeed semantically close to the original keywords.

When query expansion on the same test set was performed using expansion via words clustered using document co-occurrence data, such as described in [Sal71], precision and recall got significantly worse.

5 Conclusion

The system SEXTANT, presented here, shows that meaningful word relations can be extracted from a large corpus using imperfect syntactic analysis. It employs a knowledge-poor approach since no domain knowledge is needed in order to extract similar words. At the same time, using syntactic relations rather than coarse document co-occurrence statistics provides much finer contexts for determining similarity between words.
A few other researchers have started to explore this middle ground between simple word counting and knowledge rich approaches. Ruge [Rug91] used extracted noun phrases as her contexts, and found that analyzing which words modified which others over a large number of noun phrases gave interesting results. Hearst [Hea92] has found that frequently occurring lexical syntactic patterns, e.g. em X such as Y, Z, ..., allows the extraction of a number of hyponymic relations (Y is a type of X) over a large corpus. Like SEXTANT, these techniques are applicable to unrestricted corpora of natural language, and, more than current techniques applied to fixed resources such as dictionaries [VMd89], promise to be highly fruitful approaches as more and more text becomes available on-line.

References


