Term Extraction using a Similarity-based Approach

Diana Maynard and Sophia Ananiadou
Dept. of Computing & Mathematics
Manchester Metropolitan University
Manchester, M1 5GD, U.K.
{D.Maynard, S.Ananiadou}@doc.mmu.ac.uk

March 4, 1999

Abstract

Traditional methods of multi-word term extraction have used hybrid methods combining linguistic and statistical information. The linguistic part of these applications is often underexploited and consists of very shallow knowledge in the form of a simple syntactic filter. In most cases no interpretation of terms is undertaken and recognition does not involve distinguishing between different senses of terms, although ambiguity can be a serious problem for applications such as ontology building and machine translation. The approach described uses both statistical and linguistic information combining syntax and semantics to identify, rank and disambiguate terms. We describe a new thesaurus-based similarity measure which uses semantic information to calculate the importance of different parts of the context in relation to the term. Results show that making use of semantic information is beneficial for both theoretical and practical aspects of terminology.

1 Introduction

In recent years there has been a growing need for advances in the establishment and management of terminological resources. The vast quantities of new electronic material are simultaneously causing this requirement and enabling it to be fulfilled. In particular, new terms are constantly emerging and undergoing changes in meaning. Although it was once possible to carry out tasks such as term recognition and ontology creation manually, automatic or at least semi-automatic approaches are now required.

The increasing availability of suitable electronic texts has enabled corpus-based methods to be widely used in areas of terminology such as term recognition, but attention has been largely focused on statistical methods involving little or no understanding. Although relatively successful in practical terms, they contribute few theoretical insights. While some approaches have been of a more linguistic nature, those which involve anything more than very shallow knowledge tend to be restricted to a narrow field and are not portable to new domains.

In this paper we present a two-tiered approach to the extraction of multi-word terms. The underlying layer is corpus-based and is largely statistical, whilst the top layer exploits semantic information found both in the context and in an exterior lexical resource. This not only enables candidate terms to be identified and ranked in order of termhood, but also for disambiguation to be carried out, thereby enabling different meanings of terms to be ranked individually.

2 Fundamental Principles

2.1 Terms and Concepts

It is well known that the correct association of term and concept is important for machine translation [4], but this is also true for almost any application of computational linguistics involving specialised languages. Translation of technical texts involves finding a suitable equivalence between the source language (SL) term and the target language (TL) term. In order to establish this, equivalence between the SL and TL concept must first be found. For machine translation, the problem generally lies even deeper since the SL and TL concepts may not have a direct correspondence. So although translation appears to be simply a matter of establishing a direct term-term correspondence, it is not always quite so simple. Cognitive issues also have to be considered: a concept is not fixed in stone but can vary according to the viewpoint of an individual or a group of people. So establishing the correspondence between SL and TL concepts is not always straightforward.

Whilst we are not aiming to perform machine translation in this work, the same principles apply. We must have some comprehension of the relationship between term and concept in order to understand term variation and ambiguity, both of which are crucial to the theoretical study of terminology and to its practical applications. In order to perform term disambiguation, we need to be able to relate different occurrences of a term to different concepts, and in order to be able to account for term variation we need to be able to relate different linguistic realisations of a term to one concept. This is virtually impossible unless we already have a clear idea of what these concepts are.

2.2 Contexts

The overall picture cannot be completed without mentioning the linguistic contexts we are investigating. They can be seen as related primarily either to terms or to concepts [4]. The psychological viewpoint sees them as being related to concepts, in that the context serves to position the concept within a particular subject field. Because we view contexts as being the key to understanding a termform, we consider that it is the meaning of the term (and thereby the concept) rather than its linguistic realisation which is most relevant to the context. The alternative approach views the context from a more structural position, regarding the context as a source of collocational information and syntactic structures. In this case the context is linked to the term itself rather than to its underlying meaning. Although it is undeniable that contexts do serve this purpose, we consider this to be an incidental rather than primary function. Because we regard contexts from a semantic rather than a syntactic viewpoint, we adopt an approach that considers the context in relation to the concept rather than to the term.

According to this viewpoint, we see the concept as a *bridge* between context and term. Since we aim to illuminate the relationships not only between text and concept but also between concept and term, clearly an understanding of concepts is crucial.

2.3 Approaches to Term Recognition

Approaches to term recognition really require an in-depth understanding of the nature of terms, and thereby "not only contribute to the applications of computational linguistics but also to the theoretical foundations of terminology" [10]. Some comprehension of the behaviour of terms is thus crucial to any successful approach to automatic methods of term recognition and disambiguation.

Methods of automatic term recognition involving purely linguistic information have been fairly limited. They fall into two main categories: those which use extrinsic information and those which use intrinsic information relative to the term. The types of information used tend to be syntactic for the former, e.g. LEXTER [2], and syntactic or morphological for the latter, e.g. [1].

More recently, approaches to ATR have veered towards using both statistical and linguistic information [3],[9],[5]. Generally the main part of the algorithm is the statistical part, but shallow linguistic information is incorporated in the form of a syntactic filter which only permits certain combinations of syntactic categories to be considered as candidate terms. Little use has been made of semantic information for the purpose of term recognition, largely because, in contrast to morphological and syntactic information, it is hard both to identify and to manipulate.

2.4 Extracting Semantic Information

On the other hand, semantic information has been used fairly extensively for other purposes such as knowledge acquisition and word sense disambiguation. Because they are not used specifically for terminology, these methods tend to use general language dictionaries or thesauri. One common method is to compare the content of the dictionary definition of a word with the words in the surrounding context. Lesk [12] used an online dictionary to calculate the overlap between dictionary definitions of word senses, Smeaton [17] calculated word-word similarity between related words in a thesaurus, while Yarowsky [20] captured the meanings of word senses from semantic categories in Roget's Thesaurus.

The application of these methods to word sense disambiguation has been largely successful. There are, however, two main reasons why they are not entirely suitable for our work. Firstly, they are used for word sense rather than term sense disambiguation. Secondly, they are designed to deal with general rather than domain-specific language. Technical terms are not covered sufficiently in any general language dictionary or thesaurus, and we propose that they really require a more specialised source of information.

An alternative approach to the acquisition of semantic information involves the use of a disambiguated corpus for training. Riloff and Lehnert [16] developed an algorithm to derive relevancy cues from training texts, which they used for information extraction. Soderland et al. [18] also developed a system to identify concepts in a text, by means of linguistic features which reliably identified the conceptual content of a phrase. Grefenstette [8] adopted a hybrid, knowledge-poor approach to the automatic extraction of semantic information from large corpora, using only syntactic information. The advantage of these techniques is that they require no dictionary or thesaurus and provide information specific to the corpus. However, they do require suitable large-scale corpora.

2.5 Term sense disambiguation

In many ways, term sense disambiguation is similar to the problem of word sense disambiguation. However, it differs in two main aspects. Firstly, terms are domain-specific, which means that general language resources and techniques may not be appropriate. Secondly, the majority of technical terms consist of more than one word (the average length of NP terms is approximately 1.91 words, depending on the type of corpus). The ambiguity of multiword terms is generally not caused by different senses of the individual components of the term, but by different senses of the term as a whole. The different meanings of the term may be linked to different domains, but they may equally be present within a specific domain. Although domain-specific text is likely to contain the meanings of a term related to that domain, it does not rule out the possibility of the general meaning of that term being used as well. For example, in biochemistry, the term *complement* is used to refer to a component of blood serum, but it might equally be found in a medical text with its general meaning. Similarly, the term *drug* found in a medical text could refer to either an illegal substance or simply to a kind of medicine. Because of these differences, techniques applied to word sense disambiguation are not always appropriate for term sense disambiguation.

3 Contextual Information

3.1 NC-Value method

Our approach is built on a method for automatic term recognition called NC-Value [7]. This uses a mixture of statistical and linguistic information to rank candidate terms. We make use of the contextual information already acquired to incorporate deeper forms of linguistic knowledge, thereby improving the recognition and also performing disambiguation of terms, so that different senses of a term are distinguished and individually ranked. In Frantzi's approach, potential terms are first extracted from a corpus and ranked using the C-value method [6] based on frequency of occurrence and term length. Contextual information is then incorporated into the algorithm, in the form of weights based on the statistical characteristics of the context words.

Context words are composed of any nouns, adjectives and verbs in a fixed-size context window, either preceding or following a potential term. The context words are extracted and assigned a weight based on how frequently they appear with terms, and these weights are combined to produce a *context factor* for each term. The NC-value combines the C-value with this context factor to produce a re-ranking of the list of terms:

$$NCvalue(a) = 0.8 * Cvalue(a) + 0.2 * CF(a) \tag{1}$$

where
a is the candidate term,
$Cvalue(a)$ is the Cvalue for the candidate term,
$CF(a)$ is the context factor for the candidate term.

3.2 Identifying Relevant Contextual Information

It is well-known among terminologists that linguistic contexts provide a rich and valuable source of information. The problem lies in identifying those parts of the context which are actually relevant. Although a domain-specific corpus is used, it is clear that not all parts of the context are equally useful. Traditionally, a KWIC index is used to find a window of words surrounding candidate terms, but these contexts then have to be manually investigated - a time-consuming and laborious process.

The linguistic knowledge used in the NC-Value approach is very limited, since it is only the syntactic category of the context word which is taken into account. It considers neither any differences that might exist between the categories, nor the position in which the context word occurs with respect to the term. For example, it might be the case that nouns are more useful than verbs in predicting termhood, or that verbs preceding a term are more useful indicators than verbs following a term.

Another factor which plays an important role is that of the *meaning* of the context words. We propose two new indicators of termhood:

- context words which are themselves terms (which we call *context terms*)
- context words which are closely related in meaning to their co-occurring terms.

Our claim is that if a context word has some contribution towards the determination of a term, there should be some significant correspondence between the meaning of that context word and the meaning of the term. In other words, there should be some identifiable semantic relation between the two, which can be used to make a contribution towards the correct identification and comprehension of a term. The following two sections describe how this is achieved. The two indicators of termhood proposed above are incorporated using weights. Firstly, context terms are assigned a weight dependent on how frequently they occur with the candidate term. Secondly, a semantic weight is allocated to each candidate term based on its similarity with all its context terms.

3.3 Context Term Weight

The first parameter we consider is the status of the context word. If it is a term, we predict that it is more likely to be significant than a non-term, in other words, that terms are better indicators of other terms. This stems from the fact that terms do not tend to occur singly or randomly, but in groups, particularly in domain-specific texts.

Since we do not know in advance which context words are terms, this step can only be undertaken once we have a preliminary list of candidate terms. For this we use the top of the list of terms extracted by the C-value approach [6], since this should contain the "best" terms (or, at least, those which behave in the most term-like fashion). A context term weight (CT) is assigned to each candidate term based on how frequently it appears with a context term. It is formally described as follows:

$$CT(a) = \sum_{d \epsilon T_a} f_a(d) \tag{2}$$

where
a is the candidate term,
T_a is the set of context terms of a,
d is a word from T_a,
$f_a(d)$ is the frequency of d as a context term of a.

4 A Thesaurus-based Similarity Measure

Using a million-word corpus of eye pathology records, candidate terms are extracted using the NC-value method, along with context windows of 5 words either side of the term. Each context word and term is then tagged with its semantic type, using information provided by the UMLS Metathesaurus. Generalisations can then be made about these semantic types and the relations between them. The context words are divided into terms and non-terms. Context terms will automatically receive a higher weighting, since they are predicted to have a stronger association with other terms than context words have.

4.1 Semantic Information in UMLS

The UMLS (Unified Medical Language System) [14] is a set of knowledge sources containing information about medical terminology, organised in a hierarchical structure. Not only does it provide a classification system for the terms, but it also contains morphological, syntactic and semantic information. The Semantic Network contains additional information about the relations between the semantic classes to which the terms are assigned.

Each term and context term is tagged with its semantic type from the UMLS Metathesaurus. For example, the term *actinic keratosis* will be tagged with the semantic type *acquired abnormality*.

Relational information about the semantic types is also available. This information is generic rather than specific, i.e. details are provided about classes of terms rather than about individual terms, and do not necessarily hold for every member of that class. For example, a relationship exists between the semantic classes of *disease or syndrome* and *acquired abnormality* such that the former is a *result_of* the latter. This does not imply that any disease is the result of any acquired abnormality, but simply that there is a general relationship of this kind between the two classes, such that diseases or syndromes can be the result of acquired abnormalities. We plan to incorporate this kind of relational information into our future work.

4.2 Calculating Similarity in the UMLS Semantic Network

The similarity between context term and term is measured using the UMLS Semantic Network, which is a hierarchy of semantic types.

Our approach is modelled on techniques used in Example-Based Machine Translation (EBMT) [21], [19]. EBMT aims to select the most similar example for a given problem, by defining a set of relevant examples and comparing them with the problem in terms of context and configuration. Most of these comparison methods involve using some kind of semantic distance measure, based on the relative positions of the two items within a hierarchical network.

In its simplest form, similarity is measured by edge-counting – the shorter the distance between the words, the greater their similarity. The Most Specific Common Abstraction (MSCA) method [11] involves tracing the respective paths of the two words back up the hierarchy until a common ancestor is found, and then measuring the average distance from node to MSCA. The shorter the distance to the MSCA, the more similar the two words. The position of the MSCA within the hierarchy can also be used to measure similarity [21], [19]. The lower down in the hierarchy the MSCA, the more specific it is and therefore

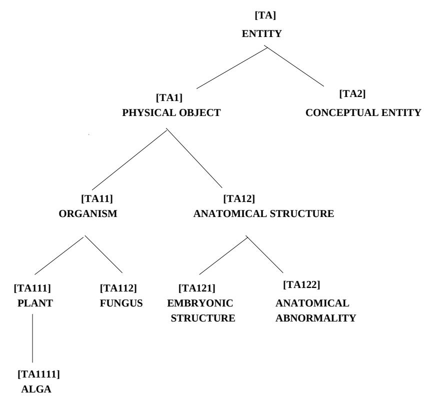

Figure 1: Fragment of the Semantic Network

the more information is shared by the two concepts, thus making them more similar. We define two weights as follows:

- *positional*: the vertical position of the nodes within the hierarchy, measured by the combined number of nodes belonging to each word
- *commonality*: essentially the horizontal distance between the two nodes, measured by the number of shared common ancestors multiplied by the number of words (usually two).

The nodes in the Semantic Network are coded such that the number of digits in the code represents the number of leaves descended from the root to that node, as shown in Figure 1. Similarity between two nodes is calculated by dividing the commonality weight by the positional weight to produce a figure between 0 and 1, 1 being the case where the two nodes are identical, and 0 being the case where there is no common ancestor (which would only occur if there were no unique root node in the hierarchy).

$$sim(w1, w2) = \frac{com(w1, w2)}{pos(w1, w2)}$$

where
com(w1,w2) is the commonality weight of word 1 and word 2
pos(w1,w2) is the positional weight of word 1 and word2.

As an example, the similarity between nodes TA111 and TA112 would be calculated as follows:-

com(TA111, TA112) = 4 * 2 = 8
pos(TA111,TA112) = 5 + 5 = 10
sim(TA111,TA112) = 8/10 = 0.8

It becomes clear that the further down in the hierarchy the two items are, the greater their similarity, which is intuitive since they are more specific: for example there would be greater similarity between *apple* and *banana* than between *fruit* and *vegetable*. It is also intuitive that the greater the horizontal and/or vertical distance between words in the network, the less similar they are.

5 Results and Evaluation

5.1 Incorporating the Weights

The measure described in the previous section calculates the similarity between each term and the "significant" context words it occurs with, i.e. the context terms. The similarity weight is then added into the original NC-Value in the following way.

The NC-Value first calculated a weight for each context word appearing with a top-ranked term, and then a context factor for each term (see section 4.1). The similarity weights are incorporated into the context factor in the same way as the weights for the context words, i.e. for each context term appearing with a candidate term, the similarity weight is multiplied by the frequency of the context term appearing with the candidate term. This is added to the original context factor. More formally, the whole context factor can now be described as:

$$CF(a) = \sum_{b \epsilon C_a} f_a(b) \cdot weight(b) + \sum_{d \epsilon T_a} f_a(d) \cdot sim(d, a) \tag{3}$$

where
a is the candidate term,
$C_a$is the set of context words of a,
b is a word from C_a,
$f_a(b)$ is the frequency of b as a context word of a,
$weight(b)$ is the weight of b as a context word,
T_a is the set of context terms of a,
d is a word from T_a,
$f_a(d)$ is the frequency of d as a context term of a,
$sim(d, a)$ is the similarity weight of d occurring with a.

5.2 Disambiguation of Terms

Since the semantic weight is calculated separately for each occurrence of a term, rather than just once for each term, this enables a distinction to be made between different usages of a term, even within a single corpus. If a term is ambiguous, it will usually have more than one semantic type, and therefore

Occurrence	No. of Context Terms	Percentage
more frequent with term	419	90%
equally frequent with either	28	6%
more frequent with non-term	17	4%

Table 1: Occurrences of context terms with terms and non-terms

List section	No. of Co-occurrences	Proportion of Co-occurrences
top	31310	76%
middle	6019	15%
bottom	3927	9%

Table 2: Occurrences of context terms with terms from each part of the list

different semantic weights can be calculated for different meanings of a term. Both intuition and experimentation showed that the meaning of a term with the highest semantic weight was most frequently the correct one in the context [13]. This also helps validate our theory that terms occur with semantically related context words.

5.3 Evaluation of the Context Term Weight

To evaluate the effectiveness of the context terms weight, we performed an experiment to compare the frequency of occurrence of each context term with terms and with non-terms. This was achieved by taking each context term individually and calculating the proportion of terms to non-terms each occurred with. The results depicted in Table 1 show that 90% of context terms appear more frequently with a term than with a non-term, and only 4% appear less frequently. The figure in the second column indicates not the total number of co-occurrences, but the number of context terms occurring in the proportion indicated by the first column (more, less or equally frequently with terms and non-terms).

Another property of context terms should be that if they appear more frequently with terms at the top of the list than terms at the bottom of the list (i.e. they occur more often with "better" terms), then they are better indicators. We therefore conducted another experiment to see how often context terms appeared with terms from each part of the list. The C-value list was again divided into 3 equal sections and the number of occurrences of context terms with terms from each section of the list was calculated. The results depicted in Table 2 confirm overwhelmingly that this is the case, 76% of all occurrences being with terms from the top of the list.

	Term	Non-Term
top set	76%	24%
middle set	56%	44%
bottom set	49%	51%

Table 3: Semantic weights of terms and non-terms

5.4 Evaluation of the Semantic Weight

One of the problems with our method of calculating similarity is that it relies on a pre-existing lexical resource. Whilst this is not a perfect solution, since there are likely to be omissions and possibly errors, there are also inherent problems with other methods such as relying solely on corpus data. Bearing in mind its innate inadequacies, we can nevertheless evaluate the expected theoretical performance of the measure by assuming completeness and concerning ourselves only with what is covered by the thesaurus.

The semantic weight is based on the premise that the more similar a context term is to the candidate term it occurs with, the better an indicator that context term is. So the higher the total semantic weight for the candidate term, the better the ranking of the term and the better the chance that the candidate term is valid. To test the performance of the semantic weight, we sorted the terms in descending order of their semantic weights and divided the list into 3, such that the top third contained the terms with the highest semantic weights and the bottom third contained those with the lowest. We then calculated the proportion of valid and non-valid terms (as determined by manual assessment) in each section of the list.

The results depicted in Table 3 reveal that in the top third of the list, 76% were terms and 24% were non-terms, whilst in the middle third, 49% were terms and 51% were non-terms. Most of the valid terms are thus contained in the top third and the fewest valid terms are contained in the bottom third. Also, the proportion of terms to non-terms in the top of the list is such that there are more terms than non-terms, whereas in the bottom of the list there are more non-terms than terms. The evaluation therefore demonstrates two things:

- More of the terms with the highest semantic weights are valid, and fewer of those with the lowest semantic weights are valid,
- More valid terms have high semantic weights than non-terms, and more non-terms have lower semantic weights than valid terms.

5.5 Comparison with Statistical Methods of Measuring Similarity

Using a pre-defined thesaurus to calculate similarity has the pitfalls that the hierarchy does not always tend to be even, especially when it has been created manually, as in the case of UMLS. So the distance between two nodes may actually reflect differing degrees of similarity depending on which section of the thesaurus they occur in. Although we take into consideration the vertical position of the nodes, this is to account for the fact that nodes lower in the hierarchy are intuitively more similar, rather than for any discrepancies in the uniformity of the ontology. Various methods of regulating this problem have been proposed, e.g. [15], [17]. In almost all similarity measures using thesauri, statistical information is either solely used or at least incorporated into the measure. However, there is no clear evidence to suggest that adding statistical information will significantly improve the measure, where the statistical part is not the primary component, and where a relatively small corpus is used as the basis for frequency information.

The methods which intuitively seem most plausible are based on information content. The information content of a node is related to its probability of occurrence in the corpus. The more frequently it appears, the more likely it is to be important in terms of conveying information, and therefore should receive a higher weighting. We performed experiments to compare two such methods with our similarity measure. The first considers the probability of the MSCA of the two terms, whilst the second considers the probability of the nodes of the terms being compared.

The probability of a node is calculated by its frequency in the corpus. Since semantic types do not appear as such in the corpus, this frequency is found by summing the frequencies of all the terms in the corpus belonging to this semantic type. To estimate the probability of a semantic type, we divide its frequency of occurrence by the total frequency of all semantic types occurring in the corpus. This can be described more formally as follows. Firstly we define the probability of a term occurring in the corpus:

$$P(t_j) = \frac{f(t_j)}{\sum_{S \epsilon T} \sum_{t_j \epsilon S} f(t_j)} \tag{4}$$

where

T is the set of semantic types occurring in the corpus
S is a semantic type belonging to T
t_j is a term belonging to a semantic type S
$f(t_j)$ is the frequency of term t_j occurring in the corpus

The probability of a semantic type occurring can then be defined as:

$$P(S_i) = \sum_{t_j \epsilon S_i} P(t_j) \tag{5}$$

where

S_i is a semantic type
t_j is a term
$P(t_j)$ is the probability of t_j occurring in the corpus

	Term	Non-Term
top set	76%	24%
middle set	57%	43 %
bottom set	49%	51%

Table 4: Experiment 1: Semantic weights of terms and non-terms

5.5.1 Experiment 1

The first experiment involves adding the probability of the MSCA to the similarity value of a term. For each combination of term and context term, the weight of the MSCA is added to the similarity weight to give a new score. The idea stems from the approach of [17] which uses the probability of the MSCA alone to calculate similarity between words. Our rationale for using the probability of the MSCA is that the MSCA is the most significant part of the hierarchy when measuring similarity, because it is the lowest part of the hierarchy (and thus the most informative part) at which the two terms are similar. Below this point, the two terms begin to differ, so any additional information encountered will not be common to both. Thus it should be beneficial to the measurement of similarity to take particular notice of this "lowest informative point" at which the similarity between the two is greatest in terms of information yielded, and thus at this point that the probability should be measured.

5.6 Experiment 2

It could also be argued that it is the probability of the individual nodes being considered that is most relevant. If a particular semantic type is statistically important, then this is likely to reflect on all the terms belonging to that semantic type. It could therefore be claimed that two terms belonging to statistically important semantic types have a greater degree of similarity than two terms which do not. Care must be taken, however, to avoid blurring the distinction between theoretical notion of similarity and its practical usage.

The second experiment involves adding the information content weight to each term being considered. So when similarity is calculated between a term and a context term, the IC weight of the node under which each is positioned is added to the similarity weight.

The experiments incorporating the idea of information content into the similarity measure were evaluated by comparing the number of terms and non-terms found in the top, middle, and bottom sections of the list of terms ordered by similarity.

The results, shown in Tables 4 and 5, are remarkably similar, and in fact differ little from the original similarity measurement shown in Table 3 and discussed earlier.

	Term	Non-Term
top set	76%	24%
middle set	59%	41%
bottom set	48%	52%

Table 5: Experiment 2: Semantic weights of terms and non-terms

6 Conclusions

As we had originally suggested, neither method was significantly better than the other, nor did either show any real improvement over our original method. This does not necessarily mean that all three methods are equivalent, however. Each of the methods does produce different results in that the values attributed to the terms differ, but this has no real impact on the overall performance, i.e. in distinguishing terms from non-terms and ranking them accordingly. The original weight and the first experiment have a similar range of values for the weights, whilst the second experiment has a greater range.

As is often the case, the criteria for evaluation is dependent on the application. The simple experiments described above show that for the purposes of our method of term ranking, any of the three similarity measures is better than none, but that there is very little difference which one is used. For other applications or with different data sets, the choice of measure might be more significant, but our purpose here is to evaluate the measures locally rather than globally, i.e. within the context of our system.

As mentioned earlier, the system could be improved by incorporating other kinds of semantic information into the measure, such as relational information not captured by the similarity measure. Information about context words is not fully exploited, in particular regarding verbs. The similarity measure itself is also lacking in that it fails to account for terms not found in the thesaurus. The results have nevertheless demonstrated the usefulness of incorporating deeper forms of linguistic information for term extraction and for its practical applications.

References

[1] S. Ananiadou. *Towards a methodology for automatic term recognition.* PhD thesis, University of Manchester, UK, 1988.

[2] D. Bourigault. Surface grammatical analysis for the extraction of terminological noun phrases. In *Proc. of COLING*, pages 977–981, 1992.

[3] B. Daille, E. Gaussier, and J.M. Langé. Towards automatic extraction of monolingual and bilingual terminology. In *Proc. of COLING 94*, pages 515–521, 1994.

[4] Robert Dubuc and Andy Lauriston. Terms and contexts. In S.E. Wright and G. Budin, editors, *Handbook of Terminology Management*, volume 1: Basic Aspects of Terminology Management, chapter 1.3.3, pages 80–87. John Benjamins, Amsterdam, 1997.

[5] K.T. Frantzi. *Automatic Recognition of Multi-Word Terms*. PhD thesis, Manchester Metropolitan University, England, 1998.

[6] K.T. Frantzi and S. Ananiadou. A hybrid approach to term recognition. In *Proceedings of NLP+IA 96*, volume 1, pages 93–98, Moncton, Canada, June 1996.

[7] K.T. Frantzi and S. Ananiadou. Automatic term recognition using contextual cues. In *Proceedings of 3rd DELOS Workshop*, Zurich, Switzerland, 1997.

[8] G. Grefenstette. *Explorations in Automatic Thesaurus Discovery*. Kluwer Academic Publishers, 1994.

[9] J.S. Justeson and S.M. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1:9–27, 1995.

[10] Kyo Kageura and Bin Umino. Methods of automatic term recognition. *Terminology*, 3(2):259–289, 1996.

[11] J.L. Kolodner. *Case-Based Reasoning*. Morgan Kaufmann Publishers Inc., San Mateo, California, 1993.

[12] M. Lesk. Automatic sense disambiguation: how to tell a pine cone from an ice cream cone. In *Proc. of the SIGDOC Conference*, pages 24–26, 1986.

[13] D. Maynard and S. Ananiadou. Identifying contextual information for term extraction. In *Proc. of 1st Workshop on Computational Terminology, Computerm '98*, Montreal, Canada, 1998.

[14] NLM, U.S. Dept. of Health and Human Services. *UMLS Knowledge Sources*, 8th edition, January 1997.

[15] P. Resnik. Disambiguating noun groupings with respect to WordNet senses. In *Proc. of 3rd Workshop on Very Large Corpora*. MIT, 1995.

[16] E. Riloff and W. Lehnert. Classifying texts using relevancy signatures. In *Proc. of AAAI.*, 1992.

[17] A. Smeaton and I. Quigley. Experiments on using semantic distances between words in image caption retrieval. In *Proc. of 19th International Conference on Research and Development in Information Retrieval*, Zurich, Switzerland, 1996.

[18] S. Soderland, D. Fisher, J. Aseltine, and W. Lehnert. CRYSTAL: Inducing a conceptual dictionary. In *Proc. of the 14th International Joint Conference on Artificial Intelligence*, 1995.

[19] E. Sumita and H. Iida. Experiments and prospects of example-based machine translation. In *Proc. of 29th Annual Meeting of the Association for Computational Linguistics*, pages 185–192, Berkeley, California, 1991.

[20] D. Yarowsky. Word-sense disambiguation using statistical models of roget's categories trained on large corpora. In *Proc. of 14th International Conference on Computational Linguistics*, pages 454–460, 1992.

[21] Gang Zhao. *Analogical Translator: Experience-Guided Transfer in Machine Translation.* PhD thesis, Dept. of Language Engineering, UMIST, Manchester, England, 1996.