Knowledge-lite extraction of multi-word units with language filters and entropy thresholds

Magnus Merkel & Mikael Andersson
Department of Computer and Information Science
Linköping University
S-581 83 Linköping, Sweden
{magme, miand}@ida.liu.se

Abstract

In this paper two approaches to knowledge-lite terminology extraction are compared, both involving language filters which are used to remove ill-formed multi-word units (MWUs). A knowledge-lite approach entails swift portability to new languages and to new domains, which is difficult to achieve if knowledge-intensive resources such as grammars, parsers, taggers and lexicons are used. The two approaches described in this paper have been applied in monolingual term extraction for translation purposes as well as in a pre-processing stage for bilingual word and MWU alignment. The implemented software has been tested for Swedish, English, German and French.

Introduction

Identifying terminology in a corpus of texts is related to the problem of identifying collocations and phrases. To produce compilations of such multi word units is not a trivial problem. Statistical methods based on frequency or measuring mutual information scores for strings of words (cf. Choueka, 1988; Smadja 1993; Nagao & Mori, 1994; Johansson, 1996; Kita et al., 1994; Dagan & Church, 1990; Shimohata et al., 1997; Yamamoto & Church, 1998; Zhou & Dapkus, 1995) can produce lists of term and phrase candidates made up of multi-word units, but there is always a possibility that units with low frequencies will be left out. Furthermore, some kind of filtering is necessary, for example by eliminating function words before, inside or at the end of the candidate (cf. Merkel et al., 1994). Other approaches involve grammatical or syntactic processing and require that the text is tagged for parts-of-speech which will make it possible to find candidates of low frequencies based on, for example, noun phrase patterns or other specified criteria (cf. Kupiec, 1993; Chen & Chen, 1994, Birn, 1997).

In this paper we will compare two knowledge-lite approaches to multi-word unit (MWU) extraction. Both are knowledge-lite in the sense that neither of the approaches requires large linguistic resources, such as language-specific grammars or lexicons. The knowledge-liteness property therefore makes the approaches possible to port to new languages with relatively little effort.

The first approach: Frequency-based MWU extraction with Language filter 1

A basic characteristic of uninterrupted collocations or multi-word units (MWUs) are that they are recurrent, domain-dependent and that the order of units often is rigid (cf. Smadja 1993). The first observation entails that *frequency* will be an important factor in order to identify well-formed MWUs in any text. The second characteristic indicates that terminology-style MWUs will be found within homogenous text types and the third observation suggests that some kind of n-gram processing techniques could form the basis for MWU extraction. The basic extraction of MWUs with a frequency threshold could be implemented in the following way:

1. Read the text into a two-dimensional array and store all words in a hash table together with their positions in the text.
2. Find all allowed MWUs up to a limit of L words by expanding candidate MWUs to the right by 1 word (in the first iteration all words taken as MWUs of length 1 are tested, with the exception of words below the frequency threshold).
3. Store the MWUs that meet the frequency threshold.
4. Go back to step 2 and expand the stored MWUs by 1 word. This is done by finding the position of the n-gram from the hash table, and extracting the next word position from the text array. The iteration then continues until no more MWUs can be added or until the limit of the longest MWU (specified as L) is reached.

When the extraction is complete all the found MWUs are examined and text positions where shorter MWUs are subsumed by longer ones are deleted, if they have the same frequency.

This is a relatively straightforward approach and is easily implemented, but the results are clearly not usable in most applications. Some extra heuristics or filtering mechanisms are necessary in order to increase the quality of the MWU output.

In Table 1, the 32 most frequent maximal MWUs extracted from a 100,000-word user's guide for a database program are listed. MWUs consisting of 3 words or more were searched for in the manner that was described above. The result is a list of maximal strings in the source text of which many are clearly unusable as terms or collocations.

Table 1. The top 32 MWUs generated from a computer program User's Guide with a pure frequency-approach

Multi-Word Unit	**Freq**	**Multi-Word Unit**	**Freq**
you want to	452	and then choose	82
, you can	392	the database window	80
for example,	327	in this chapter.	79
menu, choose	136	, click the	79
if you want	130	the edit menu	78
you can use	119	you can also	78
to create a	112	the tool bar	77
, and then	109	a form or	73
example, you	106	in design view	73
if you want to	105	choose the ok button	72
, see chapter	105	you can create	72
for example, you	102	the ok button.	71
in this chapter	100	, select the	71
the qbe grid	99	then choose the	70
form or report	94	choose the ok button.	69
, see “	88	for more information	69

When this kind of output is revised by hand, it is obvious that a majority of the MWUs removed from the original output actually are units that contain punctuation marks, that end in phrase-initial function words (such as “and”, “the”, “to”, “in”, etc.) or that contain only one parenthesis character or quotation mark.

Language filter 1

A straightforward solution to this problem is to use some kind of filtering. By adding a language filter in the process it is possible to define words that should be removed at the beginning and at the end of MWUs as well as requirements on what kinds of characters should be regarded as pairs (quotation marks, parentheses, etc).

The language filter can be tailored to specific texts, text types and languages by simply editing a word list. If the MWUs in Table 1 were to be cleaned up with the aim at locating noun phrases and prepositional phrases, the filter will contain prepositions, articles, conjunctions and frequent verbs as non-enders. A simple filter like this will, of course, not filter out all non-interesting MWUs, but it will reduce their numbers significantly.
An extract of a language filter is shown below:

```
non_starter: if when how your his her my then and or not do did

non_ender: the a an for in on is are each every want can related might click you're

non_ender: or and one also to too its save associate associated relate deleting

non_ender: if this that at you from of then your not do does did doing choose

/…/
```

The filter specifies that no MWU can begin with any of the words specified by the label "non_starter". Furthermore MWUs cannot end with any of the words given after the label "non_ender".

The language filter is then applied on-the-fly in the MWU extraction process. Whenever a maximal MWU is detected that is above the frequency threshold, words and strings specified as non-starters or non-enders are stripped from the candidate and the resulting MWU is stored in the result table.

The language filter will thus make the system filter out MWUs such as "to create a" ",and then" and "menu, choose" given an appropriate language filter.

Table 2 below shows the result of running the system on the same text as in Table 1 with the filtering mechanism activated. Of the 32 most frequent segments in the unfiltered result above, 22 have been filtered out, leaving a residue of 10.

Table 2. The top 10 segments generated with a language filter

Multi-Word Unit	Freq
in this chapter	100
the qbe grid	99
form or report	94
the database window	80
the edit menu	78
the tool bar	77
the ok button	74
in design view	73
choose the ok button	72
for more information	69

In this section the first approach to identifying terminology has been described. It entails simple n-gram extraction combined with frequency thresholds and a language filter where non-enders and non-starters can be specified. This has also been implemented in a system called Frasse-1, described in Merkel et al. (1994).

Improvements to the above approach can be done in several ways, for example by using more advanced statistics than simple frequency information. A statistical measure, often used as an indicator of collocations, is mutual information (Church and Hanks 1990). Smadja (1993) used the Dice coefficient for finding both monolingual and bilingual MWUs, and Shimohata et al. (1997) explored the possibilities of retrieving MWUs by calculating entropies for different left and right word contexts, where co-occurrence and constraints on word order will help to single out likely candidates for useful MWUs. There are studies, however, that have indicated that high frequency is a stronger indicator for retrieving MWUs than mutual information (e.g. Daille 1994), which would support the frequency approach. In the following section, we take a look at the statistical approach to use entropy thresholds to improve the performance of terminology extraction.

The second approach: Entropy-based MWU extraction with Language filter 2

To test and characterise the differences between a statistical approach, such as the one described in Shimohata et al. (1997), and the frequency-based and language filter approach described in the previous section, a second MWU extraction system was built. This system, called Frasse-2, combines the word filtering approach and constraints on entropy thresholds for the immediate contexts of MWU candidates.

Shimohata et al. (1997) describe an algorithm which is based on the observation that most MWUs appear in highly varying contexts. The words just before and after an MWU vary a great deal, while the actual MWU stays the same. The diversity of neighbouring words thus marks where MWUs start and end. The entropy value for an MWU is then a combination of the entropy values measured to the left of the MWU and to the right of it.

To measure the probability of each adjacent word ($w_1 \ldots w_n$) to a given string of words (*str*) the relative frequencies are used in the following way:

$$p(w_i \mid str) = \frac{freq(w_i)}{freq(str)}$$

The left and right entropies of a string of words, *str*, are mathematically defined as:

$$H(str) = \sum_{i=1}^{n} - p(w_i) \log p(w_i)$$

where $p(w_i)$ is the probability of seeing the word w_i adjacent to the string, and w_i are the words that do occur just before or after the string in the text.

A high entropy value signifies that the words surrounding the string, *str,* vary considerably, so strings with

$$H(str) \geq T_{entropy}$$

are accepted as MWUs. In this evaluation the same entropy threshold as adopted by Shimohata et al. (1997) is used, namely 0.75. This threshold is then combined with a frequency threshold, which indicates the minimum frequency of the string.

Language filter 2

The approach to use entropy values as a constraint for extracting MWUs was then combined with a modified version of the language filter described in the previous section. The reason for this is that when purely statistical approaches are used, a lot of meaningless recurring patterns such as “for a”, “in the”, “out of the”, etc. are extracted as likely candidates for MWUs. In Language filter 1 described in the previous section, there were in principle two ways of constraining the extraction of such MWUs:

(1) a list of words that could not start an MWU; and,
(2) a list of words that could not end an MWU.

In evaluations of the output from the first approach, it was observed that words listed as non-starters and non-enders could very well be included inside an extracted MWU, but that there was no way to force the extraction component to avoid such MWUs. For example, if the user decides that personal pronouns such as “you”, “he”, “him”, etc. should not be part of meaningful MWUs in a technical domain, then there should be a way of expressing this. Consequently, a third category of filtering entities was added: *prohibited words*, i.e., words that are forbidden to be included in MWUs. A fourth category of filtering entities was also added: *ignored words*. The latter are words that are ignored (or skipped) when entropies for surrounding contexts are measured. In some of the English applications the definite article “the” was included in this category.

The revised language filter consists of

1. a list of words that MWUs may not start with (NON-STARTERS)
2. a list of words that MWUs may not end with (NON-ENDERS)
3. a list of words, which can never be part of any MWU (PROHIBITED WORDS)
4. a list of words that are not considered part of an MWU, but do not delimit them.

In addition to these word lists, a list describing punctuation characters, which are never part of MWUs in the language in question is used.

In order to design the language filter appropriately for a new text (genre), we start out with a general language filter and add more items in the filter step-by-step to filter out. In practice, five iterations of adding non-starters, non-enders and prohibited words will suffice for a good result.

Algorithm

The algorithm when combining language filters and entropy thresholds works in the following way:

1. Read the text into a two-dimensional array and store all words in a hash table together with their positions in the text.
2. Find all allowed MWUs up to a limit of L words by expanding candidate MWUs to the right by 1 word (in the first iteration all words taken as MWUs of length 1 are tested, with the exception of words below the frequency threshold, words listed as non-starters and prohibited words).
3. For each of the allowed MWUs found in step 2, calculate the left and right context entropies and store all MWUs with an entropy value higher than the specified entropy threshold.
4. Go back to step 2 and expand the stored MWUs by 1 word. This is done by finding the position of the N-gram from the hash table, and extracting the next word position from the text array. The iteration then continues until no more MWUs can be added or until the limit of the longest MWU (specified as L) is reached.

In the second phase all the found high-entropy MWUs are examined and text positions where shorter MWUs are subsumed by longer ones are deleted. This results in a list of MWUs that meet the frequency threshold, the entropy threshold and are constrained by the language filter. The output is then listed in two formats: (1) an alphabetically sorted list of MWUs with information of the entropy value and frequency as well as the corresponding text positions; and (2) a list of MWUs sorted by size and in descending entropy order.

The differences between when Language filter 2 is activated and when it is not, are illustrated in Table 3 (the first ten entries starting with the letter L).

Table 3. Output from entropy-based MWU extraction with Language filter 2 and without filtering

With Language filter 2	Without filter
label controls	label after you
label tool	label and
last field	label controls
last name	label for
last name field	label for the
last names	label from
last page	label from a
last record	label in the
layout properties	label of
layoutforprint property	label of the

In the next section, the two approaches to MWU extraction will be compared and evaluated in more detail.

Comparison of "Frequency/Filter1" and "Entropy/Filter2"

What are the results in terms of output data and efficiency if the two different approaches to extraction of MWUs are compared? In the first approach, only frequency data and a language filter were used to constrain the different multi-word units (frequency/filter1). In the second approach, frequency thresholds and a modified version of the language filter were combined with a statistical entropy measure that calculated the probabilities for adjacent strings to be considered as well-formed MWUs (entropy/filter2).

The first approach was first implemented in 1994 and is described in (Merkel et al. 1994). The second approach has been implemented more recently and actually subsumes the first approach if the entropy threshold is deactivated and if the class of *prohibited words* is not used.

In the following evaluation, the same text has been used for both approaches, namely the English Microsoft Access User's Guide. The text contains 179,631 words. The language filters that were used in both cases were identical, with the exception of 10 prohibited words which were only included in the second approach (entropy/filter2). The extraction process was run in two configurations, one with the frequency threshold set to 4 occurrences and one configuration with a frequency threshold of 2. For the entropy-based approach, the entropy threshold was set to 0.75. The minimum size of the MWU was 2 words for both the configurations.

Recall cannot be easily measured as there is no reliable way to determine how many well-formed (or practically usable) MWUs there are in a given text, but comparisons can be made between the approaches and thus relative recall can be measured. Precision has been tested on all the output of all the MWUs starting with one randomly selected letter, namely *L*. Here the MWUs have been judged as either "good" or "bad" in a broad sense. A "good MWU" is defined as a well-formed lexical unit (term, phrase or collocation) and those which do not belong to this class are simply judged as "bad".

The evaluated sample is relatively small and restricted to one text, so any definite conclusions are hard to make, but the results certainly show a strong tendency.

Table 4. Comparison of frequency-based and entropy-based configurations

Configuration		**Frequency threshold**	**Total extracted MWUs**	**MWUs in sample (letter L)**	**Good**	**Bad**	**Pre-cision**
Approach 1	Freq.+Lang.filter1	4	2655	55	43	12	82.08
	Freq.+Lang.filter1	2	7516	193	114	79	59.06
Approach 2	Entropy +Lang.filter2	4	953	29	29	0	100.00
	Entropy +Lang.filter2	2	1880	49	49	0	100.00

The strengths and weaknesses of the two approaches are obvious from Table 4. Using only frequency and the first language filter (without specifying “prohibited words) as constraints will produce a higher number of MWUs, but of lower quality. When entropy thresholds are used on the other hand fewer MWUs will be extracted (low recall), but with 100 per cent accuracy.

To illustrate the difference between the two approaches, a listing of the first 20 MWUs starting with the letter L extracted by the respective configuration (frequency threshold equal to 2) are shown in Table 5 below. The MWUs considered to be ill-formed are italicised.

Table 5. Examples of MWUs extracted by the frequency-based and the entropy-based approaches. In the entropy-based approach the “prohibited words” class was specified.

Frequency-based with Language filter 1		**Entropy-based with Language filter 2**	
MWU	**Freq.**	**MWU**	**H(str)**
label control	3	label controls	3.04
label controls	9	label tool	3.04
label from a control, and later	*3*	last field	2.93
label of any size, by clicking	*2*	last name	7.24
label of the text box	2	last name and first name fields	3.16
label that microsoft access sizes as you type, by clicking	*2*	last name field	4.64
label to start and then typing the text for the label	*2*	last names	1.83
label to start, dragging the pointer until the label is the size you want, and then typing the text in the label	*2*	last page	1.83
label to the control	*2*	last record	4.9
label tool in the toolbox	3	layout and formatting elements	3.16
label tool to create a label, the label is freestanding -	*2*	layoutforprint property	3
label tool	8	levels of subforms	2.32
label, and resize the section	*2*	limit records	3.71
label, double-click	*6*	limittolist property	4.47
label, it’s no longer attached to the text box	*2*	line item	2.89
label, text box	*2*	line of text	2.24
labels in the page header	3	lines and rectangles	3.84
labels into the page header	*2*	link data	2.71
last dialog box	8	linkchildfields property	1.94
last dialog box, click the finish button to display	*3*	linked object	5.14

Language filter 2 (where it is possible to specify “prohibited words) will actually filter out many of the extracted MWUs that are unwanted. For example, if the words “a”, “the”, “that”, “you”, “it’s” and “to” are included in the prohibited words. On the other hand, because of the lower recall of the entropy-based approach, this configuration will not extract some of the well-formed MWUs found by the frequency-based configuration, primarily because these MWUs do not get over the entropy threshold. It is worth pointing out that if prohibited words are used, the MWUs extracted will naturally be shorter, but it does not necessarily mean that the longer MWUs are missed altogether. Language filter 2 will “break” an MWU as soon as a prohibited word appears, but the strings before and after that prohibited word may very well turn out to be acceptable MWUs. Consider an MWU such as “linking fields in the main report” extracted by the frequency-based approach with Language filter 1. Although the second approach could not detect this particular MWU, two of the subsumed strings are passed as acceptable MWUs by the entropy-based approach: “linking fields”, and “main report”. In some applications, like word and phrase alignment, it is more likely that successful links can be made between the shorter subsumed MWUs and their corresponding target units, as these will have higher frequencies than the longer MWU. However, each approach has its unique strength. For automatic processing, when there are few or no resources available for manual revision of MWU data, the entropy-based configuration with Language filter 2, would ensure that the MWUs being extracted are of high quality. Furthermore, the entropy-based approach is better at detecting more terminology-style MWUs as too long MWUs are avoided unless they pass the language filter and entropy thresholds. To illustrate that entropy/filter2-approach does not avoid longer MWUs altogether consider the following six-word MWUs that this configuration extracted:

northwind employee sales by country report

employee sales by country 2 report

sales by sale amount 2 report

summary of sales by year report

last name and first name fields

northwind alphabetical list of products report

order id and order date fields

international section of windows control panel

The data illustrated in Table 4 above do however not provide information on what the individual contributions of the entropy thresholds and the more constrained language filter actually are. By applying Language filter 2 to the frequency-based processing and the less constraining Language filter 1 to the entropy-based approach, the following results were obtained.

Table 6. Comparison of applying Language filter 2 to the frequency-based approach and Language filter 1 to the entropy-based approach.

Configuration	**Frequency threshold**	**Total extracted MWUs**	**MWUs in sample (letter L)**	**Good**	**Bad**	**Precision**
Freq.+Lang.filter2	2	2180	56	52	4	92.86
Entropy +Lang.filter1	2	2,850	65	58	7	89.23

The use of prohibited words in Language filter 2 together with the pure frequency-based approach actually constrains recall significantly. Only 2180 MWUs were extracted with this configuration and with a relatively high precision. Similarly, by not filtering out MWUs containing prohibited words in the entropy-based approach, recall was increased (2,850 MWUs), but at the cost of lower precision (89.23 per cent). These figures illustrate that the use of a more constrained Language filter 2

(prohibited words) will increase precision at the expense of recall in both the frequency- and the entropy-based approaches. To summarise, the entropy-based approach to the extraction of terminology-style MWUs come out better as far as precision is concerned, even when the frequency threshold is set as low as 2, and is therefore the preferred choice when no human revision of the output can be done, for example in bilingual word and MWU alignment.

For other purposes, when a human terminologist has the possibility to revise the more imprecise data from the frequency-based approach and delete unwanted MWUs, the approaches that result in higher recall might be the preferred solution. Another option is to use both approaches; first the entropy/filter2-based configuration to acquire a high-quality core of MWUs and then add the output from the frequency-based extraction after human revision.

In practical tests, the current implementation is efficient as far as speed is concerned. The implementation has been done in Perl, both for the Sun Solaris platform and the Windows NT platform. Using a 500 MHz Pentium III PC with 256 MB of RAM, Frasse-2 with a frequency threshold of 2 and a 0.75 entropy threshold) takes just under 5 minutes when run on a 179,631 word text.

The main advantage for either of the approaches described in this paper is that they do not require any linguistic information, apart from what the user would like to insert in the language filters. This means that adaptations to new (western-European) languages are easily done (the implementations have been applied on English, Swedish, German and French texts). The use of shallow data and statistics is therefore its strength but also its weakness, as it is difficult to generalise the language filters by using word categories or grammar rules. With parts-of-speech tagged texts, it would be possible to specify the language filters in grammatical terms, and apply them together with the statistical machinery. Furthermore, Shimohata et al. (1997) and Smadja (1993) have shown how significant n-grams, i.e. uninterrupted MWUs, can be expanded to interrupted phrasal templates, such as “For more information on …, refer to the … manual …”. This is one interesting feature that could be included in future versions.

Applications

The approaches to MWU extraction presented in this paper have been used as a module in the Linköping Word Aligner (LWA), cf. Ahrenberg et al. (1998). The MWU extraction module is here used as pre-processors of the sentence-aligned bitext to produce candidate MWUs for the source text and the target text respectively. For the bilingual alignment of MWUs, it is also required that non-terminological MWUs such as multi-word adverbials, etc. are detected. This is handled by a separate language-specific list of general phrases and collocations that will be retrieved if they are present in the text in question. It should therefore be noted that the MWU extraction approaches described in the paper is focused on rigid content phrases, but will not be able to find non-terminological phrases such as *after all, in any case, among other things, once and for all,* etc. Such MWUs are instead considered to be general and listed separately. The extracted terminological MWUs and the general MWUs are then used as input for the word alignment program as potential tokens for alignment.

Acknowledgements

This work has been supported by the Swedish National Board for Industrial and Technical Development and the Swedish Council for Research in the Humanities and Social Sciences.

References

Ahrenberg, L., M. Andersson and M. Merkel (1998). A Simple Hybrid Aligner for Generating Lexical Correspondences in Parallel Texts. In *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics* , Montréal, Canada, 10-14 August 1998: 29-35.

Birn, J. (1997). A Noun Phrase Extractor for Swedish. Report. Lingsoft Inc. Helsinki.

Choueka, Y. (1988). Looking for needles in a haystack. In *Proceedings from RIAO-88*, User-oriented Content-based Text and Image Handling: 609-623.

Church, K.W. and P. Hanks (1990). "Word association norms, mutual information and lexicography." *Computational Linguistics* 16(1): 22-29.

Dagan, I. and K.W. Church (1997). "Termight: Coordinating Humans and Machines in Bilingual Terminology Translation." *Machine Translation 12(1-2):* 89-107.

Daille, B. (1994). Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. *Proceedings of the workshop The Balancing Act - Combining Symbolic and Statistical Approaches to Language*:29-36.

Johansson, C. (1996). Good Bigrams. In *Proceedings from the 16th International Conference on Computational Linguistics (COLING-96).* Copenhagen: 592-597.

Kita, K., T. Omoto Y. Yano and Y. Kato. (1994). Application of Corpora in Second Language Learning - The Problem of Collocational Knowledge Acquisition. In *Proceedings of the Second Annual Workshop on Very Large Corpora (WVLC-2).* Kyoto: 43-56.

Kupiec, J. (1993). An Algorithm for Finding Noun Phrase Correspondences in Bilingual Corpora. In *Proceedings of the 31st Annual Meeting of the Association of Computational Linguistics (ACL-93):* 17-22.

Merkel, M., B. Nilsson and L. Ahrenberg. (1994). A Phrase-Retrieval System Based on Recurrence. In *Proceedings of the Second Annual Workshop on Very Large Corpora (WVLC-2).* Kyoto: 99-108.

Nagao, M. and S. Mori (1994). A New Method of N-gram Statistics for Large Number of n and Automatic Extraction of Words and Phrases from Large Text Data of Japanese.In *Proceedings from the 15th International Conference on Computational Linguistics (Coling-94).* Kyoto: 611-615.

Shimohata, S., T. Sugio, and J. Nagata (1997). Retrieving Collocations by Co-occurrences and Word Order Constraints. In *Proceeding of the 35th Conference of the Association for Computational Linguistics (ACL'97),* Madrid: 476-481.

Smadja, F. (1993). "Retrieving Collocations from Text: Xtract." Computational Linguistics 19(1): 143-177.

Yamamoto, M. and K.W. Church (1998). Using Suffix Arrays to Compute Term Frequency and Document Frequency for all Substrings in a Corpus. In *Proceedings of the Sixth Workshop on Very Large Corpora*. E. Charniak. Montreal: 28-37.

Zhou, J. and P. Dapkus (1995). Automatic Suggestion of Significant Terms for a Predefined Topic. In *Proceedings of the Third Workshop on Very Large Corpora*. Cambridge: 131-147.