Recognizing Acronyms and their Definitions in Swedish Medical Texts

Dimitrios Kokkinakis[‡] **and Dana Dannélls**[§]

Göteborgs Universitet

[‡]Department of Swedish Language, Språkdata, [§]Department of Linguistics

Box 200, SE-405 30 Göteborg

E-mail: dimitrios.kokkinakis@svenska.gu.se, cl2ddoyt@cling.gu.se

Abstract

This paper addresses the task of recognizing acronym-definition pairs in Swedish (medical) texts as well as the compilation of a freely available sample of such manually annotated pairs. A material suitable not only for supervised learning experiments, but also as a testbed for the evaluation of the quality of future acronym-definition recognition systems. There are a number of approaches to the identification described in the literature, particularly within the biomedical domain, but none of those addresses the variation and complexity exhibited in a language other than English. This is realized by the fact that we can have a mixture of two languages in the same document and/or sentence, i.e. Swedish and English; that Swedish is a compound language that significantly deteriorates the performance of previous approaches (without adaptations) and, most importantly, the fact that there is a large variation of possible acronym-definition permutations realized in the analysed corpora, a variation that is usually ignored in previous studies.

1. Introduction

Lexical acquisition plays an essential part in getting natural language processing systems to increase their performance in real world tasks and various forms of electronic dictionaries are the essential tools for understanding language in many technical fields, such as biomedicine. A characteristic within the biomedical field is the exponential growth of new volumes of information, and consequently biomedical terminology, particularly new abbreviations and acronyms (the latter is usually considered a subset of the previous). An important piece of information crucial for keeping up to date lexical ontologies and dictionaries; components essential for a number of tasks, such as information retrieval, information extraction, text normalization and text mining. Chang et al. (2002) report that 64,262 new abbreviations were introduced in MEDLINE (http://medline.cos.com/) during 2001, an average of 1 new abbreviation every 5-10 articles.

Acronym identification is the task of processing an arbitrary text in order to annotate and/or extract a pair of strings from it. An acronym is a result of taking a phrase and shortening it into a new form. The new form is a string, usually a short mixed sequence of characters, possibly Roman/Arabic numbers or other alphanumerics. The phrase is an expanded word form which provides the definition or exemplification of the acronym. The first may precede or follow the later. This paper addresses the problem of recognizing acronym-definition pairs in Swedish (medical) texts. In connection to this we have compiled a sample of manually annotated pairs, freely available for research. A material suitable not only for supervised learning experiments, but also as a test bed for the evaluation of the quality of future acronym-definition recognition systems.

For instance, in the sentence, taken form the MEDLEX annotated sample; (Kokkinakis, 2006; see Section 3): *"The new approach is based upon the <acronym link="1">OMG</acronym>'s (<definition id="1">Object Management Group</definition>) <definition id="2">Model Driven Architecture </definition> <acronym link="2">MDA</acronym>) framework […]*", there are two acronyms annotated, 'OMG' and 'MDA' linked to their two expansions, 'Object Management Group' and 'Model Driven Architecture'.

Several approaches have been proposed for automatic acronym recognition and extraction in the literature, particularly within the biomedical domain, but none of those addresses the variation and complexity exhibited in a language other than English. The most common methods include pattern-matching techniques and machine learning algorithms. The implementation presented in this paper applies a rule-based algorithm to process and automatically detect different forms of acronym-definition pairs, while different machine learning algorithms are tested by using the acronym pair candidates recognized by the rule-based component, represented as feature vectors for the supervised machine learning experiments.

This paper starts by a brief investigation of some of the previous approaches to the problem of acronym-definition recognition (Section 2) and continues with a description of the manually inspected and annotated sample, the MEDLEX Acronym-Definitions Data (Section 3). Section 4 provides a description of the implementation of the acronym-definition recognition system and discusses some language-specific difficulties of the task. Section 5 presents the obtained results and the evaluation based on the MEDLEX annotated sample. Finally, conclusions and future work end the paper.

2. Background

The task of automatically extracting acronym-definition pairs from biomedical literature has been studied, almost exclusively for English, using technologies from Natural Language Processing (NLP). This section presents some of the most significant and influential approaches for the

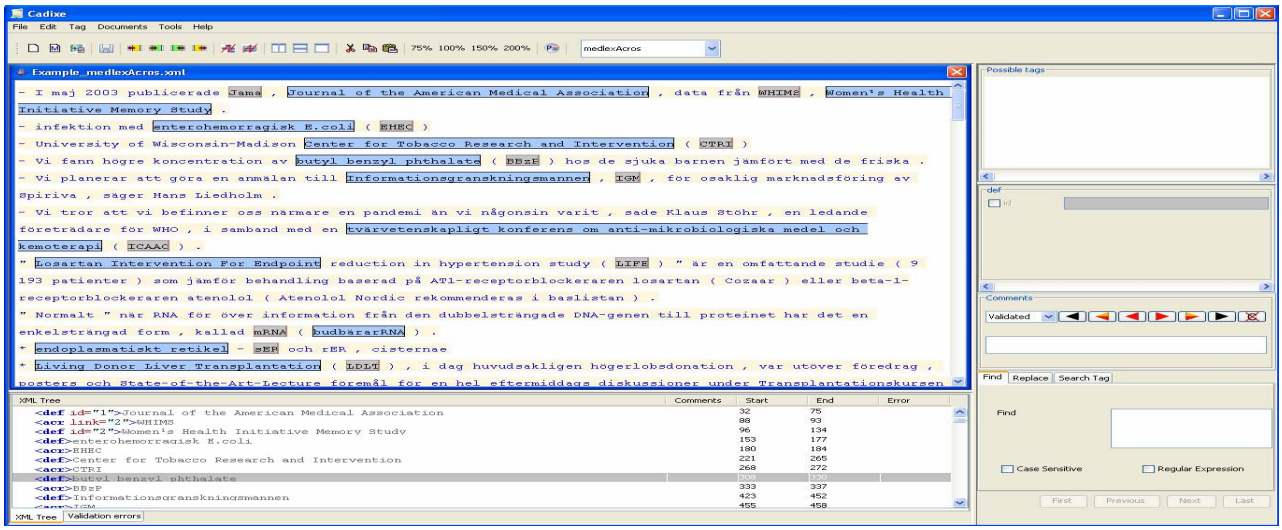

Figure (1). Interactive annotation with the CADIXE XML-editor

task. Taghva and Gilbreth (1999) present the *Acronyms Finding Program* (AFP), based on pattern matching. Their program seeks for acronym candidates which appear as upper case words. They calculate a heuristic score for each competing definition by classifying words into: (1) stop words (”the”, ”and”); (2) hyphenated words; (3) normal words (words that don’t fall into any of the above categories) and (4) the acronyms themselves (since an acronym can sometimes be a part of the definition). The AFP utilizes the Longest Common Subsequence (LCS) algorithm (Hunt and Szymanski, 1977) to find all possible alignments of the acronym to the text, followed by simple scoring rules which are based on matches. The performance reported from their experiment is recall 86% and precision 98%. An alternative approach to the AFP was presented by Yeates (1999). In his program, *Three Letters Acronyms* (TLA), he uses more complex methods and general heuristics to match characters of the acronym candidate with letters in the definition string. The results achieved by TLA were 91% recall of and 68% precision.

Another approach recognizes that the alignment between an acronym and its definition often follows a set of patterns (Park and Byrd, 2001 and Larkey et al., 2000). Pattern-based methods use strong constraints to limit the number of acronyms respectively definitions recognized and ensure reasonable precision. Nadeau and Turney (2005) present a machine learning approach that uses weak constraints to reduce the search space of the acronym candidates and the definition candidates; they reached a recall of 89% and precision of 88%. Schwartz and Hearst (2003) present a simple algorithm for extracting abbreviations from biomedical text. The algorithm extracts acronym candidates, assuming that either the acronym or the definition occurs between parentheses and by giving some restrictions for the definition candidate such as length and capital letter initialization. When an acronym candidate is found the algorithm scans the words to the right and left side of the found acronym and tries to match the shortest definition that matches the letters in the acronym. Their approach is based on previous work (Pustejovsky et al., 2001), they achieved recall of 82% at precision of 96%. It should be emphasized that the common characteristic of all the previous approaches in the surveyed literature is the use of parentheses as well as that the acronyms are in upper case, as indication for the acronym pairs, see the table 1 in Nadeau and Turney’s (2005). These limitations have many drawbacks since it excludes the acronym-definition candidates which do not occur within parentheses and thereby do not provide a complete coverage for all the acronyms formation (cf. Table 1).

3. The Medlex Acronym-Definitions Data

We have manually annotated using simple XML markaup a set of 861 acronym-definition pairs. The set was extracted from Swedish medical texts, the MEDLEX corpus, (Kokkinakis, 2006), and it is tokenized[1]. The material has been annotated using the CADIXE XML Annotation Editor (see http://caderige.imag.fr/) (Figure 1). For the majority of the cases in the sample, there exists one acronym-definition pair per sentence, but there are cases where two or more pairs can be found. Since Swedish is a compounding language we also provide a version of the data set where all compounds have been automatically segmented. An example of such annotation is given below (‘||’ marks a compound segmentation point): *”<definition>Apolipo||protein E</definition> (<acronym>ApoE</acronym>) and geno||typning vid utredning av hyper||lipid||emi och athero||skleros […]”*. Compound segmentation can in many cases provide the right means for making easier the recognition of a definition. For example, “***L****äkemedels||****i****ndustri||****f****ör-eningen (LIF)*”, The Swedish Association of the Pharmaceutical Industry.

Table (1) shows the distribution of the definition-acronym pairs in the annotated corpus sample. Some examples include:

A=D:	AVNRT = AV-nodal reentrytakykardi
A-D:	ACE - Angiotensin Converting Enzyme
A(D):	PAI-1 (plasminogen activator inhibitor 1)
D(A):	reumatoid artrit (RA) .
A, D,:	tPA, tissue-type plasminogen activator,
D, A,:	C-reaktivt protein, CRP,

[1] The longest definition found in the sample was for the acronym “RESTORE”, in the context “*RESTORE (Reconstructive Endoventricular Surgery returning Torsion Original Radius Elliptical shape to the left ventricle) är en grupp hjärtkirurger…*”.

pattern	#occurr.	%
D (A)	570	66,2%
D, A ,	122	14,2%
A (D)	49	5,7%
D, A .	44	5,1%
A, D ,	12	1,4%
“D” (A)	9	1%
A = D	6	<1%
A - D	4	<1%
Rest	45	5,2%

Table 1. Distribution of definitions (D) and acronyms (A) (e.g. “A (D)” means an Acronym followed by a Definition within parenthesis.

4. Implementation

A drawback of previous systems is that they only seek for acronym candidates which appear in uppercase, thus the acronym candidate “*amyotrofisk lateralskleros (als)*”, i.e. “Amyotrophic Lateral Sclerosis”, wouldn’t have been considered by systems such as AFP. Moreover, most of the previous systems would have failed to match acronyms that consist of two characters, such as “per rectum (PR)”, a frequent structure in the Swedish annotated material. Considering also that we want to recognize all possible variation of (Swedish) acronym-definition pairs it is practical to use pattern-based techniques to extract relevant information of which a suitable set can give a valid representation of the different acronym pairs and thus making non-trivial prediction on new data. The method presented in this section is inspired by Nadeau and Turney’s work and is based on the algorithm described by Schwartz and Hearst. Our method starts by using a pattern-based algorithm that has the advantage of recognizing acronym-definition patterns even outside parentheses and continues with the machine learning component. The algorithm matches acronyms with their related definitions based on a pre-defined set of heuristics that limits the search for acronym-definition candidates.

4.1 Acronym and Definition Candidates

Each word in the text file is considered as an *acronym candidate* if it is a string of alphabetic, numeric and/or includes special characters such as '-' and '/'. The string becomes a valid acronym candidate if:

- The string contains at least two characters, and
- The string is not in the list of rejected words[2], and
- The string contains at least one capital letter, or the string’s first or last character is a lower case letter or numeric.

Each *definition candidate* string is passed through a number of heuristics of which all are necessary in conjunction:

- At least one letter of the words in the string matches the letter in the acronym.
- The string doesn't contain a colon, semi-colon, question mark or exclamation mark.
- The maximum length of the string is min(|A|+5,|A|*2)[3].
- The string doesn't contain only upper case letters.

These heuristics allows acronym-definition candidates such as: “*autoimmun kronisk hepatit, (aiKH)*” [4] ; “*Hemocult II, (H-II)*”; ”*atopiskt eksem/dermatit syndrom , (aeds)*”; “*in vitro-fertilisering/embryo transfer,(IVF/ET)*” and “*human Metapneumovirus, (HCoV-NL63)*”.

4.2 Matching Acronym-Definition

The process of matching an acronym with its definition depends on their appearance in the text. According to the algorithm there exist two matching possibilities:

(1) *Parentheses matching*. The algorithm extracts acronym-definition candidates which correspond to one of the following patterns (cf. Schwartz and Hearst, 2003):

a) definition (acronym)
b) acronym (definition).

(2) *Non parentheses matching*. The algorithm extracts acronym-definition candidates which are not enclosed in parentheses.

The algorithm scans the text for an acronym candidate that satisfies the conditions described in Section 4.1. When an acronym is found, the algorithm searches the words surrounding the acronym for a definition candidate string according to the heuristics in the same section. The search space for the definition candidate string is limited to 4 words * |A|.

The next step is to choose the correct substring of the definition candidate for the acronym candidate. This is done by reducing the definition candidate string as follows: the algorithm searches for identical characters between the acronym and the definition starting from the end of both strings and succeeds in finding a correct substring for the acronym candidate if it satisfies the following conditions:

a) At least one character in the acronym string matches with a character in the substring of the definition
b) The first character in the acronym string matches the first character of the leftmost word in the definition substring, ignoring upper/lower case letters

An example of a potential acronym-definition pair that was (correctly) failed during this process is: “*peritoneal dials University Utrecht, FG*”, since there was no letter match is found to the string *FG*.

4.3 Machine Learning Approach

To test and compare different supervised learning algorithms, the Tilburg Memory-Based Learner, TiMBL, was used; (Daelemens et al., 2004). Feature vectors were calculated to describe the acronym-definition pairs. Ten numeric features were chosen: (1) the acronym or the definition is between parentheses (0-false, 1-true), (2) the definition appears before the acronym (0-false, 1-true), (3) the distance in words between the acronym and the definition, (4) the number of characters in the acronym, (5) the number of characters in the definition, (6) the number of lower case letters in the acronym, (7) the number of lower case letters in the definition, (8) the number of upper case letters in the acronym, (9) the number of upper case letters in the definition and (10) the number of words in the definition. The 11th feature is the class to predict: true candidate (+), false candidate (-). An example of the

[2] The rejected word list contains frequent acronyms which appear in the corpus without their definition, e.g. 'USA’, ‘EU’.
[3] |A| is the acronym’s length (cf. Park and Byrd, 2001).
[4] Including variations: “autoimmun kronisk hepatit, aiKH”, “aiKH, (autoimmun kronisk hepatit)” and “aiKH, autoimmun kronisk hepatit”.

acronym-definition pair "*vCJD, variant CJD*" represented as a feature vector is: 0,1,1,4,11,1,7,3,3,2,+.

5. Results and Evaluation

The rule-based component was evaluated on unannotated instances from the MEDLEX Corpus, using the standard precision/recall metrics. The results obtained were 92% precision and 72% recall. The algorithm recognized 671 acronym-pairs of which 619 were correctly identified. A closer look at the 52 incorrect pairs showed that the algorithm failed to make a correct match when: (i) characters are skipped in the acronym string such as "*Institutionen för fysiologi och farmakologi, (FYFA*)"; (ii) a character in the acronym string don't match any character in the definition string, such as "*glycol alginate lösning, (PGA)*"; (iii) letters in the definition that were not in the acronym, due to a mixture of a Swedish definition with an English-based acronym, "*datortomografi , CT*", where CT stands for "Computer Tomography"; and (iv) mixture of words with (roman) numerals, "*Usher typ III (USH3)*" or other phenomena "*39-item Parkinson's Disease Questionnaire (PDQ 39)*". The algorithm also failed to find three letters acronyms which consist of lower-case letters and do not appear within parentheses, such as "*apolipoproteinerna, apo, […]*".

The machine learning component used the acronym-definition pairs recognized by the rule-based algorithm as the training data. The 671 pairs were presented as feature vectors according to the features described in Section 4.3. The material was divided into two data files; 80% training and 20% test data. Four different algorithms were used to create models. These algorithms were: IB1, IGTREE, TRIBL and TRIBL2. The results obtained are given in Table 2.

algorithm	precision	recall	f-score
IB1	90.6 %	97.1 %	93.7 %
IGTREE	95.4 %	97.2 %	96.3 %
TRIBL	92.0 %	96.3 %	94.1 %
TRIBL2	92.8 %	96.3 %	94.5 %

Table 2: Memory-Based algorithm results.

6. Conclusion and Future Work

We have outlined our work for the creation of a manually annotated sample of acronyms and their expanded forms from Swedish medical corpora and presented a method for acronym-definition recognition based on this material. Our pattern-based algorithm was designed to deal with the variety of Swedish acronyms that are seen in authentic Swedish medical texts. The algorithm has the advantage of recognizing acronym-definition pairs which are not only indicated by parentheses. It utilizes predefined heuristics to find and extract acronym-definition pairs with different patterns; a strategy which has proven to be a suitable for this task and that can be further improved.

One of the drawbacks of the algorithm is that it tries to match characters starting at the end of both the acronym and the definition strings, using a backward search algorithm. To increase recall it is necessary to combine forward search algorithm to match characters starting at the leftmost side of the strings. Moreover, different algorithms such as the LCS algorithm will be appropriate to combine with the existing code. The algorithm should be evaluated on other data before conclusions can be drawn and it will be interesting to test the algorithm on a different corpus. Although not tested, we speculate that our method will perform just as good for other languages such as English, as well as for other domains. In the near future we plan to add new features to the method such as database lookup, part-of-speech tagging and/or noun-phrase chunking. The performance of the machine learning experiments can be further improved by modifying the input settings e.g test different feature weighting schemes. One advantage for applying machine learning techniques is that decisions which are made by a certain learning scheme, based on one set of examples, could later be applied to any given text with unseen acronym pairs. Machine learning can also help to select the heuristics that are most appropriate for matching acronyms with definitions (as suggested by Yeats, 1999). On-going work aims to improve the rule-based method and combine it in a better way with the machine learning component while experimenting with new sets of features.

7. Acknowledgements

This work has been partially supported by the *Semantic Interoperability and Data Mining in Biomedicine* - NoE 507505. We would like to thank Gilles Bisson for allowing us to use the CADIXE editor.

8. References

Chang J.T., Schütze H., Russ B. and Altman R.B. (2002). *Creating an Online Dictionary of Abbreviations from MEDLINE*. JAMIA PrePrint.

Daelemans W., Zavrel J., van der Sloot K. and van den Bosch A. (2004). *TiMBL: Tilburg Memory Based Learner, version 5.1, Reference Guide*. ILK Technical Report 04-02, Available from http://ilk.uvt.nl/downloads/pub/papers/ilk0402.pdf'

Hunt J.W. and Szymanski T.G. (1977). *A Fast Algorithm for Computing Longest Common Subsequences*. Com. of the ACM. 20(5): 350–353.

Kokkinakis D. (2006). *Collection, Encoding and Linguistic Processing of a Swedish Medical Corpus: The MEDLEX Experience*. Proc. of the 5th Language Resources and Evaluation Conference (LREC). Genoa, Italy.

Larkey L. et al. (2000). *Acrophile: An Automated Acronym Extractor and Server.* Proceedings of the ACM Digital Libraries Conference. Pp. 205-214. San Antonio, Texas

Nadeau D. and Turney P. (2005). *A Supervised Learning Approach to Acronym Identification*. Proc. of the 18th Conference of the Canadian Society for Computational Studies of Intelligence LNCS 3501. Pp. 319-329. Canada.

Park Y. and Byrd R.J. (2001). *Hybrid Text Mining for Finding Abbreviations and their Definitions*. Proc. of the Empirical Methods in Natural Language Processing. Pittsburgh, PA.

Pustejovsky J. et al. (2001). *Automation Extraction of Acronym-Meaning Pairs from MEDLINE Databases*. Medinfo 2001;10 (Pt 1): 371-375.

Schwartz A. and Hearst M. (2003). *A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Texts*. Proc. of the Pacific Symposium on Biocomputing. Hawaii.

Taghva K. and Gilbreth J. (1999). Recognizing Acronyms and their Definitions. *Journal on Document Analysis and Recognition (IJDAR)*. 1:191-198. Springer-Verlag.

Yeates S. (1999). *Automatic Extraction of Acronyms from Text*. Proc. of the 3rd New Zealand Computer Science Research Students' Conference. University of Waikato..