A Web-Mining Approach to Disambiguate Biomedical Acronym Expansions

Mathieu Roche
LIRMM - UMR 5506, CNRS
Univ. Montpellier 2,
34392 Montpellier Cedex 5 - France

Violaine Prince
LIRMM - UMR 5506, CNRS
Univ. Montpellier 2,
34392 Montpellier Cedex 5 - France

Keywords: web-mining, text-mining, natural language processing, BioNLP, named entities recognition, acronym, quality measures

Received: September 25, 2008

Named Entities Recognition (NER) has become one of the major issues in Information Retrieval (IR), knowledge extraction, and document classification. This paper addresses a particular case of NER, acronym expansion (or definition) when this expansion does not exist in the document using the acronym. Since acronyms may obviously expand into several distinct sets of words, this paper provides nine quality measures of the relevant definition prediction based on mutual information (MI), cubic MI (MI3), and Dice's coefficient. A combinaison of these statistical measures with the cosine approach is proposed. Experiments have been run on biomedical domain where acronyms are numerous. The results on our biomedical corpus showed that the proposed measures were accurate devices to predict relevant definitions.

Povzetek: Predstavljene so metode spletnega preiskovanja dvoumnih akronimov v domeni biomedicinskih baz.

1 Introduction

Named Entities Recognition (NER) has become one of the major issues in Natural Language Processing (NLP). The state-of-the-art literature in NER mostly focuses on proper names, temporal information, specific expressions in some technical or scientific fields for domain ontologies building, and so forth. A lot of work has been done on the subject, among which on acronyms, seen as particular named entities. Acronyms are very widely used in every type of text, and therefore have to be considered as a research issue as linguistic objects and as named entities.

An **acronym** is composed from the first letters of a set of words, written in uppercase style. This set of words is generally frequently addressed, which explains the need for a shortcut. It is also a specific multiword expression, such as 'Named Entities Recognition', abbreviated into NER, sometimes completely domain dependent (as NER or NLP are). In some cases, acronyms become proper names referring to countries or companies (like USA or IBM). However, most of the time, acronyms are domain or period dependent. They are contracted forms of multiword expressions where words might belong to the common language. As contracted forms, they might be highly ambiguous since they are created out of words first letters. For instance, NER, the acronym we use for `Named Entities Recognition` might also represent `Nippon Electrical Resources` or `Natural Environment Restoration`. An **expansion** (called **definition** too) is the set of words that defines the acronym.

In all cases, an acronym behaves like a named entity. However, the intrinsic ambiguity in most acronyms enhances the difficulty of finding which exact entity is referred by this artificial name. Literature has been addressing acronym building and expansion (see section 'related work') when the acronym definition is given in the text. However, choosing the right expansion for a given acronym in a given document, if no previous definition has been provided in the text, is an issue definitely belonging to NER, and not yet exhaustively tackled. The difficulty in acronym disambiguation is to automatically choose, as an expansion, the most appropriate set of words. This article tries to deal with this issue by offering a **quality measure** for each candidate expansion. In this context, let us name a a given acronym. For every a which expansion is lacking in a document d, we consider a list of n possible expansions for a: $a^1...a^n$. For instance, if NER is the acronym at stake, we could have NER^1 = `Named Entities Recognition`, NER^2 =`Nippon Electrical Resources`, and NER^3 =`Natural Environment Restoration`. Some web resources exist for providing acronym definitions as http://www.sigles.net/

or specialized biomedicine resources given by http://www.nactem.ac.uk/software/acromine/. In the experiments of this paper we have focused on biomedical data (18) because this domain uses a lot of polysemic acronyms.

The aim of our approach is to determine k ($k \in [1, n]$) such that a^k is the relevant expansion of a in the document d. To make such a choice, we provide different quality measures which relie on Web resources.

The presentation is structured as following: section 2 discusses the output of the related literature, section 3 focuses on the quality measure $AcroDef$, where context and web resources are essential characteristics to be taken into account. The section 4 extends the Turney's measures that we call $IADef$ measures. Section 5 gives an example of the nine quality measures based on $AcroDef$ and $IADef$ measures. Section 7 describes some experiments about $AcroDef$ and $IADef$ measures on biomedical domain. Finally conclusion and future work are suggested in section 8.

2 Related work

Among the several existing methods for acronyms and acronyms expansion extraction in the literature, we present here some significant works. First, acronyms detection within texts is an issue by itself. It involves recognizing a character chain as an acronym and not as an unknown or misspelled word. Most acronyms detecting methods rely on using specific linguistic markers.

Yates' method (28) involves the following steps: First, separating sentences by segments using specific markers (brackets, points) as frontiers.

For instance, the sentence:

```
The NER (Named Entity Recognition) system is
presented.
```

will become

```
The NER | Named Entity Recognition | system is
presented |
```

The second step compares each word of each segment with the preceding and following segments. In our example, the following comparisons are performed:

- `The` with `Named Entity Recognition`
- `NER` with `Named Entity Recognition`
- `Named` with `The NER`
- `Entity` with `The NER`
- and so forth...

Then the couples acronym/expansion are tested. The candidates acronym/definition are accepted if the acronym characters correspond to the first letters of the potential definitions words. In our example, the pair 'NER/Named Entity Recognition' is a good acronym/expansion candidate. The last step uses specific heuristics to select the relevant candidates. These heuristics rely on the fact that acronyms length is smaller than their expansion length, that they appear in upper case, and that long expansions of acronyms tend to use 'stop-words' such as determiners, prepositions, suffixes and so forth. In our example, the pair 'NER/Named Entity Recognition' is valid according to these heuristics.

Other works (2; 10) use similar methods based on the presence of markers associated to linguistic and/or statistical heuristics. For example, some recent works as (15) use statistical measurements from terminology extraction field. Okazaki and Ananiadou apply the C-value measure (7; 14) initially used to extract terminology. This one favors a candidate term that not appears often in a longer term. For instance, in a specialized corpus (Ophthalmology), the authors found the irrelevant term 'soft contact' while the frequent and longer term 'soft contact lens' is relevant. The advantage of the measure proposed by (15) is the independence of the characters alignment (actually, a lot of acronyms/definitions are relevant while the letters are in a different order as 'AW / water activity').

Other approaches based on supervised learning methods consist in selecting relevant expansions. In (27), the authors use the SVM approach (Support Vector Machine) with features based on acronyms/expansions informations (length, presence of special characters, context, etc). The work of (24) presents a comparative study of the main approaches (supervised learning methods, rules-based approaches) by combining domain-knowledge.

Our method is closer than Word Sense Disambiguation (WSD) approaches summarized in (13). A part of these WSD approaches uses machine-learning techniques to learn a classifier from labeled training sets (22; 9). In our case, we consider our method like unsupervised. But our system based on statistical measures and web-mining techniques differs with "bag of words" approaches described in (13). Note that our method will be combined with approaches of the literature to disambiguate definitions of biomedical domain (see section 6).

Larkey *et al.*'s method (10) uses a search engine to enhance an initial corpus of Web pages useful for acronym detection. To do so, starting from a list of given acronyms, queries are built and submitted to the AltaVista search engine.[1] Queries results are Web pages which URLs are explored, and eventually added to the corpus. Our method shares with (10) the usage of the Web. However, we do not look for existing expansions in text since we try to determine a possible expansion that would be lacking in the text where the acronym is detected. From that point of view, we are closer to works like Turney's (25), which are not

[1] http://www.altavista.com/

specifically about acronyms but which use the Web to define a ranking function. The algorithm PMI-IR (Pointwise Mutual Information and Information Retrieval) described in (25) queries the Web via the AltaVista search engine to determine appropriate synonyms to a given query. For a given word, noted $word$, PMI-IR chooses a synonym among a given list. These selected terms, noted $choice_i$, $i \in [1, n]$, correspond to the TOEFL questions. The aim is to compute the $choice_i$ synonym that gives the better score. To obtain scores, PMI-IR uses several measures based on the proportion of documents where both terms are present. Turney's formula is given below (1): It is one of the basic measures used in (25). It is inspired from Mutual Information described in (3).

$$score(\ choice_i\ ) = \frac{nb(\ word\ NEAR\ choice_i\ )}{nb(\ choice_i\ )} \quad (1)$$

- $nb(x)$ computes the number of documents containing the word x,
- $NEAR$ (used in the 'advanced research' field of AltaVista) is an operator that precises if two words are present in a 10 words wide window.

With this formula (1), the proportion of documents containing both $word$ and $choice_i$ (within a 10 words window) is calculated, and compared with the number of documents containing the word $choice_i$. The higher this proportion is, the more $word$ and $choice_i$ are seen as synonyms. More sophisticated formulas have also been applied: They take into account the existence of negation in the 10 words windows. For instance, the words 'big' and 'small' are not synonyms if, in a given window, a negation associated to one of these two words has been detected, which is likely to happen, since they are antonyms (opposite meanings).

To enhance relevance to the document, our $AcroDef$ approach described in section 3 calculates the dependency between the words composing the possible expansions in order to rank them. In that sense, it is close to Daille's approach (4) which uses statistical measures to rank terms. Also, as defended in next section, we use other quality measures and attempt to relate as much as possible to the context, in order to significatively enhance basic measures.

3 Defining the $AcroDef$ measure

Several quality measures in the literature (8) are based on ranking function. They are brought out of various fields: Association rules detection, terminology extraction, and so forth.

To determine the expansion of an acronym starting from a list of co-occurrences of set of words, our aim is to provide a relevance ranking of this set using statistical measures. The most appropriate definition has to be placed at the top of the list by the $AcroDef$ (section 3) and $IADef$ (section 4) measures described in the following sections.

3.1 Basic $AcroDef$ measure based on Dice's coefficient

In this paper, the $AcroDef$ measure based on the Dice's coefficient is described. Other statistical measures like Mutual Information (MI) (3) and Cubic MI (26; 5) can be used. They are presented in the subsection 3.2.

Dice's Coefficient and Mutual Information are simple and effective because they use weak knowledge. Actually, they are based on a number of examples (in our case, the number of pages provided by a search engine and queries with the words of expansions) without the need to determine the counter-examples. Indeed, the counter-examples (used by a lot of quality measures (8)) are often more difficult to find in an unsupervised context based on statistical data from the Web.

The Dice's coefficient (21) used by our basic $AcroDef$ measure computes a sort of relationship between the words composing what is called a **co-occurrence**. This measure is defined by the following formula:

$$D(x, y) = \frac{2 \times P(x, y)}{P(x) + P(y)} \quad (2)$$

For instance, with the acronym 'IR', x might represent the word 'Information' and y the word 'Retrieval'. It might also be a pair such as 'International' and 'Relations'.

Formula (2) leads directly to formula (3).[2]

$$Dice(x, y) = \frac{2 \times nb(x, y)}{nb(x) + nb(y)} \quad (3)$$

Petrovic *et al.* (17) present an extension of the original Dice formula to three elements. In a natural way, we could extend this approach to n elements as follows:

$$Dice(x_1, ..., x_n) = \frac{n \times nb(x_1, ..., x_n)}{nb(x_1) + ... + nb(x_n)} \quad (4)$$

Since our work, like many others, relies on Web resources, the nb function used in the preceding measures represents the number of pages provided by the search engine Exalead (http://www.exalead.fr/). The choice of Exalead has been determined by the fact that this search engine uses the NEAR function like the Turney's approach (formula (1)). This function will be used in other quality measures (i.e. $IADef$ measures) described in section 4.

Starting from the n extended Dice's formula (4), and using statistics provided by search engines we propose the basic $AcroDef$ measure (formula (5)).

[2] by writing $P(x) = \frac{nb(x)}{nb\_total}$, $P(y) = \frac{nb(y)}{nb\_total}$, $P(x, y) = \frac{nb(x,y)}{nb\_total}$

$$AcroDef_{Dice}(a^j) = \frac{\left|\{a_i^j | a_i^j \notin M_{stop}\}_{i\in[1,n]}\right| \times nb(\bigcap_{i=1}^{n} a_i^j)}{\sum_{i=1}^{n} nb(a_i^j | a_i^j \notin M_{stop})} \quad \text{where } n \geq 2 \tag{5}$$

- $\bigcap_{i=1}^{n} a_i^j$ represents the set of words a_i^j ($i \in [1,n]$) seen as a string (using *brackets* with Exalead and illustrated as follows: "$a_1^j ... a_n^j$"). Then an important point of this formula is that the order of the words a_i^j is taken into account to calculate their dependency.
- M_{stop} is a set of stop-words (prepositions, determiners, etc). Then the pages containing only these words are not taken into account.
- $|.|$ represents the number of words of the set.

We used the acronym 'IR' as a basic example. With $a =$`IR`, two definitions are available:

a^1: `Information Retrieval`
and a^2: `International Relations`

Let us precise that the resulting pages numbers with both definitions are:

- $a_1^1 \cap a_2^1 =$ `Information`$\cap$`Retrieval`: $366,508$ resulting pages
- $a_1^2 \cap a_2^2 =$ `International` $\cap$ `Relations`: $1,021,054$ resulting pages

The obtained values with the $AcroDef$ formula (5) are:
$AcroDef_{Dice}(\texttt{IR}^1) = \frac{2 \times nb(\texttt{Information} \cap \texttt{Retrieval})}{nb(\texttt{Information}) + nb(\texttt{Retrieval})} = \frac{2 \times 366508}{513072210 + 3202458} = 0.0014$

$AcroDef_{Dice}(\texttt{IR}^2) = \frac{2 \times nb(\texttt{International} \cap \texttt{Relations})}{nb(\texttt{International}) + nb(\texttt{Relations})} = \frac{2 \times 1021054}{234463128 + 47716188} = 0.0072$

Practically, the first result comes back to submitting the three following queries to Exalead: `"Information Retrieval"` (`Information` $\cap$ `Retrieval`), `Information` and `Retrieval`.

In languages, many noun phrases contain stop-words such as determiners or prepositions, and thus, several acronym expansions will be composed of such elements. So, when the definition of an acronym contains a stop-word, it is neglected in the formula denominator. In English, stop-words are scarce, but sometimes appear in the acronym: Part-Of-Speech in often referred to as POS in computational and general linguistics. It designates the grammatical/lexical category to which the word belongs (verb, noun, etc). The preposition 'of' has given its first letter to the acronym, probably because it simplifies the acronym pronunciation.

3.2 Basic $AcroDef$ measure based on mutual information (MI and MI3)

We can use other statistical measures to calculate the dependancy between the words x and y: Mutual Information (MI) – formula (6) – and Cubic MI – formula (7). These measures are described in (20).

$$MI(x,y) = \frac{nb(x,y)}{nb(x) \times nb(y)} \tag{6}$$

$$MI3(x,y) = \frac{nb(x,y)^3}{nb(x) \times nb(y)} \tag{7}$$

Let us note that MI tends to extract rare and specific co-occurrences according to (23). Vivaldi *et al.* have estimated that the Cubic MI (MI3) was the best behaving measure (26). Then MI3 is used in several works related to terminology (26) and complex named entities extraction in texts (5).

Then we can use these formulas ((6) and (7)) in order to define other $AcroDef$ measures, respectively based on MI and Cubic MI. $AcroDef_{MI}$ and $AcroDef_{MI3}$ are given as follows:

$$AcroDef_{MI}(a^j) = \frac{nb(\bigcap_{i=1}^{n} a_i^j)}{\prod_{i=1}^{n} nb(a_i^j | a_i^j \notin M_{stop})} \quad \text{where } n \geq 2 \tag{8}$$

$$AcroDef_{MI3}(a^j) = \frac{nb(\bigcap_{i=1}^{n} a_i^j)^3}{\prod_{i=1}^{n} nb(a_i^j | a_i^j \notin M_{stop})} \quad \text{where } n \geq 2 \tag{9}$$

These measures enable to provide different experiment comparisons in section 7.

These basic formulas ((5), (8), (9)) do not take the context into account. This is a severe liability. Therefore, next subsection details a measure that relies on context to define a more relevant expansion choice for a given acronym.

3.3 Contextual $AcroDef$

In this paper, context is defined as a set of significant words present in the page where the acronym to expand is found. Of course, other definitions of the context notions have to be considered as extensions to this preliminary approach. However, even in this restricted point of view, several operational expressions of the context could be used:

- The n most frequent words (excepting stop words);
- The n most frequent proper names;
- The n most rare words;
- POS tags (1) or terminological information present in the surroundings of the considered item.

A combination of these expressions could also be envisaged. The experiments presented in this article (section 7) use a context represented by the most frequent words, and give satisfying results. In a sequel work, we plan to define the context with a richer set of information, namely, linguistic knowledge (lexical, syntactic, semantic) as the WSD (Word Sense Disambiguation) approaches (13) do.

Adding contextual information to $AcroDef$ (formula (5)) leads to formula (10). The principle underlying this formula is to apply statistical measures on a set of words of a given domain. So, the goal is not to count the dependency between the words of an acronym definition and those of the context, but to restrict the searching space. This restriction is a requirement for the word dependency computation (and not otherwise). The formula is written as follows:

$$AcroDef_{Dice}(a^j) = \frac{\left|\{a_i^j \text{ AND } C | a_i^j \notin M_{stop}\}_{i\in[1,n]}\right| \times nb(\bigcap_{i=1}^{n} a_i^j \text{ AND } C)}{\sum_{i=1}^{n} nb(a_i^j \text{ AND } C | a_i^j \notin M_{stop})}$$

$$\text{where } n \geq 2 \quad (10)$$

In this formula, a_i^j AND C represents the pages containing the word a_i^j with all the words of the context C. For this we use the *AND* operator of Exalead. Our experiments presented in (20) show that the use of a context improves the results. If we consider our example a =IR with its two possible expansions (Information Retrieval and International Relations), the favored definition with $AcroDef$ is still International Relations with the 0.0072 value against the 0.0014 value for Information Retrieval. If we take as a context the following $C = \{\text{corpus}\}$ then we have:

$$AcroDef_{Dice}(\text{IR}^1) = \frac{2 \times nb(\text{Information} \cap \text{Retrieval AND corpus})}{nb(\text{Information AND corpus}) + nb(\text{Retrieval AND corpus})} = \frac{2\times 19270}{2079155+55253} = 0.0181$$

$$AcroDef_{Dice}(\text{IR}^2) = \frac{2 \times nb(\text{International} \cap \text{Relations AND corpus})}{nb(\text{International AND corpus}) + nb(\text{Relations AND corpus})} = \frac{2\times 5075}{1020428+281055} = 0.0078$$

In this example the relevant expansion choosen (i.e. having the best score) is the first definition (i.e. Information Retrieval).

We can add the context C in the basic measures based on MI and MI3 measures (formulas (8) and (9)) presented in section 3.2. $AcroDef_{MI}$ and $AcroDef_{MI3}$ using the context are given as follows:

$$AcroDef_{MI}(a^j) = \frac{nb(\bigcap_{i=1}^{n} a_i^j \text{ AND } C)}{\prod_{i=1}^{n} nb(a_i^j \text{ AND } C | a_i^j \notin M_{stop})} \quad \text{where } n \geq 2 \quad (11)$$

$$AcroDef_{MI3}(a^j) = \frac{nb(\bigcap_{i=1}^{n} a_i^j \text{ AND } C)^3}{\prod_{i=1}^{n} nb(a_i^j \text{ AND } C | a_i^j \notin M_{stop})} \quad \text{where } n \geq 2 \quad (12)$$

These different measures are language independent. They are tested in section 7, dedicated to experimentating $AcroDef$ on 'real' biomedical data.

4 $IADef$ measure: an expansion of Turney's measure

In the previous sections we presented the $AcroDef$ measures, that compute dependency between words forming the expansions. Such measures help choosing the relevant definitions. This approach is close to the work based on terminology extraction techniques (ranking of extracted terms) (4).

The $IADef$ (**I**ndependency between **A**cronyms and **Def**initions) measures presented in this section are closer to Turney's method described in section 2. $IADef$ computes the dependency between acronyms and definitions.

4.1 Basic Turney's measure for the acronym disambiguisation

P. Turney (25) has provided a formula (13) calculating the dependency between an acronym a and a candidate definition $\bigcap_{i=1}^{n} a_i^j$ (using *brackets* with Exalead). This formula is based on the standard measure of Mutual Information (MI).[3]

$$IADef_{MI}^{And}(\, a^j \,) = \frac{nb(\, a \text{ AND } \bigcap_{i=1}^{n} a_i^j \,)}{nb(\, \bigcap_{i=1}^{n} a_i^j \,)} \quad (13)$$

For instance, $nb(\, a \text{ AND } \bigcap_{i=1}^{n} a_i^j \,)$ with a = IR and $\bigcap_{i=1}^{2} a_i^j$ = Information $\cap$ Retrieval calculates the number of pages returned by the query IR AND "Information Retrieval". Thus, we compute the number of times where the terms IR and 'Information Retrieval' are present in the same page.

To be more precise in the calculation of the dependency between both words a (e.g. 'IR') and $\bigcap_{i=1}^{n} a_i^j$ (e.g. 'Information Retrieval'), we can compute the number of pages where the words are in a same window using the NEAR operator of Exalead. Actually, this operator requires that both words are within 16 words of each other.[4] The formula (14) calculates this dependency:

[3] In this formula, the constant $\frac{1}{nb(a)}$ is not taken into account because it does not change the order of expansions given by the statistical measure.

[4] Informations about the use of the NEAR operator of Exalead : http://www.searchengineshowdown.com/blog/exalead/ or http://moritzlegalinformation.blogspot.com/ 2006_06_01_archive.html

$$IADef_{MI}^{Near}(\ a^j\ ) = \frac{nb(\ a \text{ NEAR } \bigcap_{i=1}^{n} a_i^j\ )}{nb(\ \bigcap_{i=1}^{n} a_i^j\ )} \quad (14)$$

4.2 Turney's measure based on different statistical measures

Turney's Measure can be extended using other statistical criteria that have been described in section 3: Cubic Mutual Information and Dice's coefficient.

Cubic Mutual Information gives a greater weight in the score of the formula's numerator that calculates the dependency between terms (acronym and definition). Formulas (15) and (16) describe such measures. They use the functions AND (formula (15)) and NEAR (formula (16)) of the search engine Exalead.

$$IADef_{MI3}^{And}(\ a^j\ ) = \frac{nb(\ a \text{ AND } \bigcap_{i=1}^{n} a_i^j\ )^3}{nb(\ \bigcap_{i=1}^{n} a_i^j\ )} \quad (15)$$

$$IADef_{MI3}^{Near}(\ a^j\ ) = \frac{nb(\ a \text{ NEAR } \bigcap_{i=1}^{n} a_i^j\ )^3}{nb(\ \bigcap_{i=1}^{n} a_i^j\ )} \quad (16)$$

In addition to conventional measures such as MI and MI3, we propose to use the Dice's coefficient applied to the $IADef$ measure (formulas (17) and (18)).

$$IADef_{Dice}^{And}(\ a^j\ ) = \frac{2 \times nb(\ a \text{ AND } \bigcap_{i=1}^{n} a_i^j\ )}{nb(\ a\ ) + nb(\ \bigcap_{i=1}^{n} a_i^j\ )} \quad (17)$$

$$IADef_{Dice}^{Near}(\ a^j\ ) = \frac{2 \times nb(\ a \text{ NEAR } \bigcap_{i=1}^{n} a_i^j\ )}{nb(\ a\ ) + nb(\ \bigcap_{i=1}^{n} a_i^j\ )} \quad (18)$$

4.3 Contextual $IADef$

Like $AcroDef$ measures, we can take into account a context C (see section 3.3) with these new measures described in section 4.

Then we add a context C (using the 'AND' operator) to the queries of the formulas (13), (14), (15), (16), (17), and (18). This context enables to enhance the original measure of P. Turney (25).

5 Applying those measures: a few examples

This section provides examples of the nine quality measurements, applied to the acronym 'IR'. Actually, with these measures, we calculate the obtained score with the possible expansion 'Information Retrieval':

- $AcroDef_{Dice}$ – formula (10): $\frac{2 \times nb(\texttt{Information} \cap \texttt{Retrieval})}{nb(\texttt{Information}) + nb(\texttt{Retrieval})}$
- $AcroDef_{MI}$ – formula (11): $\frac{nb(\texttt{Information} \cap \texttt{Retrieval})}{nb(\texttt{Information}) \times nb(\texttt{Retrieval})}$
- $AcroDef_{MI3}$ – formula (12): $\frac{nb(\texttt{Information} \cap \texttt{Retrieval})^3}{nb(\texttt{Information}) \times nb(\texttt{Retrieval})}$
- $IADef_{Dice}^{And}$ – formula (17): $\frac{2 \times nb(\texttt{IR AND (Information} \cap \texttt{Retrieval))}}{nb(\texttt{IR}) + nb(\texttt{Information} \cap \texttt{Retrieval})}$
- $IADef_{Dice}^{Near}$ – formula (18): $\frac{2 \times nb(\texttt{IR NEAR (Information} \cap \texttt{Retrieval))}}{nb(\texttt{IR}) + nb(\texttt{Information} \cap \texttt{Retrieval})}$
- $IADef_{MI}^{And}$ – formula (13): $\frac{nb(\texttt{IR AND (Information} \cap \texttt{Retrieval))}}{nb(\texttt{Information} \cap \texttt{Retrieval})}$
- $IADef_{MI}^{Near}$ – formula (14): $\frac{nb(\texttt{IR NEAR (Information} \cap \texttt{Retrieval))}}{nb(\texttt{Information} \cap \texttt{Retrieval})}$
- $IADef_{MI3}^{And}$ – formula (15): $\frac{nb(\texttt{IR AND (Information} \cap \texttt{Retrieval))}^3}{nb(\texttt{Information} \cap \texttt{Retrieval})}$
- $IADef_{MI3}^{Near}$ – formula (16): $\frac{nb(\texttt{IR NEAR (Information} \cap \texttt{Retrieval))}^3}{nb(\texttt{Information} \cap \texttt{Retrieval})}$

Of course, we add a context C with these basic measures. The section 7 gives the results of these nine quality measures.

6 A hybrid approach

The context used by the $AcroDef$ and $IADef$ measures is very small (often less than three words). Therefore, results are less attractive than with methods using a large context based on "bags of words" representations.

The work presented in this section proposes a hybrid method relying on a vector representation and $AcroDef$/$IADef$ measures, in order to improve results of the precision (see section 7.4). This hybrid measure is called $IAcos$.

6.1 A vector space model to disambiguate biomedical definitions

Expanding ambiguous biomedical abbreviations is an asset. Thus, several Word Sense Disambiguation (WSD) techniques use a Vector Space Model (16; 22) to represent various possibilities. The hybrid method represents the context of an abbreviation to disambiguate, by a vector which elements are the occurrences of its close words. With such a representation, several machine learning techniques can be applied (13), particularly in the biomedical domain: SVM (22; 9), Naive Bayes (22; 9), Decision trees (9), and so forth. These techniques can use a richer representation based on linguistic features like Part-of-Speech

tags, bigrams (two consecutive words that occur together) (9), or semantic knowledge like MeSH (22), UMLS (12). In this paper, domain-knowledge is not addressed, since our approach is not specific to the sole biomedical field; It can be adapted to other domains and languages (20).

Here, an unsupervised approach is applied, a technique rather seldom developed in the biomedical disambiguisation literature. Among the few who have investigated such a process, one of the most representative is the work of (16), which consists in building contexts (bag of words) in order to predict the relevant meaning of an acronym. This context is provided by three types of corpora (i.e. Unrestricted Web, Medline abstract, Mayo Clinic). For each definition, the process developed by (16) allows to generate a context vector of lexical items and their frequency (using a window of ± 20 words). The last step of the process is based on the computation of the vectors closeness. The largest cosine is selected in order to choose the adapted definition (meaning) of an acronym in a given context. Our hybrid approach, detailed in sections 6.2 and 7.4 uses this principle associated with the $IADef$ and $AcroDef$ measures.

6.2 Our $IAcos$ method

In the first step, $IAcos$ consists in building a context (1) for the candidate definitions based on a web corpus and (2) for the document where the acronym must be defined. Like (16)'s approach, our method consists in selecting the definition having the best cosine value.

In the second step, only the definitions which are in the first positions with the $AcroDef$ or $IADef$ measures are selected. Selection aims at improving the quality of the relevant expansions returned by the system. This technique takes into account both informations returned by the cosine and web-mining methods ($AcroDef$ and $IADef$). The formula (19) gives the $IAcos$ measure.

$$IAcos_i = \max_j \{cos(d, \text{context}(a^j)) \quad (19)$$
$$/\ a^j \text{ is in the } i \text{ first definitions}$$
$$\text{returned by } IADef \text{ or } AcroDef\}$$

In this formula (19):

- d represents the vector of the document where the acronym have to be defined.
- context(a^j) is the context of candidate-definition a^j.

The method to build the context and the experimental protocol are detailed in section 7.4.

7 Experiments

7.1 Experimental protocol

In our experiments, we have focused on a classification of biological data definitions, provided by the Acromine application.[5] For any given acronym in this area, Acromine provides a list of its possible expansions. 102 pairs acronym/definitions have been randomly extracted from Acromine, which provided, for each tested item, from 4 to 6 possible definitions. The acronyms we study can be either two, three or four character strings. For instance, JA, PKD, and ABCD are possible acronyms, and for the latter, its definitions are described in the table 1. As one can see, it might range from medicine to biochemistry, dentistry, etc.

polycystic kidney disease
protein kinase D
proliferative kidney disease
paroxysmal kinesigenic dyskinesia
pyruvate kinase deficiency

Table 1: Extract of some definitions of the PKD acronym in biomedicine.

For each of these pairs, articles abstracts have been extracted from the specialized bibliographical data base Medline,[6] containing acronyms and their expansions. This base contains 204 documents (two documents per couple acronym/expansion, manually extracted). The goal of this experiment is to determine whether, for each document, the definition could be correctly predicted by classifying the candidate definitions with our quality measures. The distribution of the 204 documents according of the number of plausible candidate expansions for acronyms is given in the table 2. This table shows we need $12\times6+120\times5+72\times4 = 960$ expansions to test.

This experiment has needed the run of **7340 queries**:[7]

- Calculation of the 6 $IADef$ measures: $IADef$ measures require 2×960 queries for the numerator (with the AND and NEAR operators) and 2×960 for the denominator (for Dice measure): 3840 queries.
- Calculation of the 3 $AcroDef$ measures: $AcroDef$ requires 960 queries for the numerator and 2540 for the denominator (the number of queries for the denominator depends of the number of words of each expansion): 3500 queries.

Nb of documents	Nb of possible expansions per document
12	6
120	5
72	4

Table 2: Number of possible acronym definitions for the 204 documents.

[5] http://www.nactem.ac.uk/software/acromine/
[6] http://www.ncbi.nlm.nih.gov/PubMed/
[7] Experiments conduced in august 2009.

7.2 Results of $AcroDef$ and $IADef$ measures

Table 3 presents the results of these experiments. For each of the $AcroDef$ and $IADef$ measures:

- The first column value is the number of times where the correct definition has been given, as a first item,
- the second column value corresponds to the number of times it has been predicted among the two first definitions (ranks 1 and 2 according to the measure classification),
- and the third value corresponds to the number of times it appears among the first three.

Ranks	1	1 or 2	1, 2, or 3
$AcroDef_{Dice}$	73 (35.8%)	127 (62.3%)	161 (78.9%)
$AcroDef_{MI}$	62 (30.4%)	111 (54.4%)	149 (73.0%)
$AcroDef_{MI3}$	72 (35.3%)	118 (57.8%)	165 (80.9%)
$IADef^{And}_{Dice}$	111 (**54.4%**)	150 (**73.5%**)	174 (**85.3%**)
$IADef^{Near}_{Dice}$	104 (51.0%)	142 (69.6%)	174 (85.3%)
$IADef^{And}_{MI}$	94 (46.1%)	139 (68.1%)	169 (82.8%)
$IADef^{Near}_{MI}$	90 (44.1%)	137 (67.1%)	170 (83.3%)
$IADef^{And}_{MI3}$	104 (51.0%)	145 (71.1%)	174 (85.3%)
$IADef^{Near}_{MI3}$	102 (50.0%)	146 (71.6%)	170 (83.3%)

Table 3: Number of correct definitions based onthe expansions ranks provided by the statistical measure (Medline Abstracts)

Measure	Sum
$AcroDef_{Dice}$	470
$AcroDef_{MI}$	516
$AcroDef_{MI3}$	481
$IADef^{And}_{Dice}$	**389**
$IADef^{Near}_{Dice}$	403
$IADef^{And}_{MI}$	422
$IADef^{Near}_{MI}$	424
$IADef^{And}_{MI3}$	401
$IADef^{Near}_{MI3}$	405

Table 4: Sums of the Ranks of Relevant Definitions.

Experiments have been led with a one-word context only, i.e., the most frequent word in each document. Working on a specialized domain, queries with more than one word have null pages results with a general search engine such as Exalead.

Table 3 shows some important facts, that might provide answers to the following questions and meet some of the assigned goals:

- **Which are the best quality measures?**

Table 3 shows that $IADef$ measures give better results than $AcroDef$ measures. It seems that the calculation of the acronyms and expansions dependency is more relevant than the dependency between the expansions words. Another important conclusion is that the $IADef$ *measure based on Dice's coefficient gives the best result*. This one is best than the result obtained with the original Turney's measure (quality measures based on MI). Note that MI3 provides good results too (close to Dice's coefficient).

Table 3 shows that the performance of AND and NEAR operators is very close. This result differs from the study presented in (25). It can be explained by the specificity of acronyms usage. Indeed, acronyms and their expansions are often very close, in the same sentence, in the documents returned by the search engine. Thus, there are only little differences in the documents returned by AND and NEAR operators.

In order to determine more precisely the quality of these measures, we have computed the sum of the ranks of relevant definitions. The best measure is the one that has the smallest sum. This method, while evaluating rank functions, is equivalent to approaches based on ROC (Receiver Operating Characteristics) curves and to the calculus of surfaces under them (6; 19). Therefore, Table 4 confirms that $IADef_{Dice}$ behaves as the best measure in specialized documents belonging to biomedicine. Also, every $IADef$ measures has a better rank (smaller sum) than the best $AcroDef$ measure: The 'worst' $IADef$ result, 424, is above $AcroDef_{Dice}$, the best one among $AcroDef$ results, with 470. Note that Dice's coefficient enhances both measures results.

- **Significance of results:**
  $AcroDef^{And}_{Dice}$ hits the good definition on rank 1 in 54.4% of the cases. This is significantly better than a random prediction, which scores 22%. We calculated this random prediction as such: 1 chance over 4 to put the relevant definition as the first one in 72 cases, 1 over 5 in 120 cases, and 1 over 6 in 12 cases, which are the number of documents with respectively 4, 5, and 6 possible definitions (in Table 2).

- **Restricting the definition space:**
  The high predictive values for the first three definitions ranked by $IADef$ measures restrict the search space. It is useless to go down further in the list, and in the 204 documents where more than 4 definitions occur, it would be efficient to restrict to the first three chosen by our measures, and give the user the opportunity of choosing the best one. Further, they might be close definitions as we will show it in a deeper study of the data content.

7.3 Data properties

The retrieved definitions has led us to formulate some comments. Among the difficulties encountered in NLP research in the biomedical domain, the fact that several terms could address the same or very similar concepts is a very classical issue. For instance, when we retrieved the acronym ZO we had the following definitions: `zonula occludens`, `zona occludens`, `zonulae occludentes`. As one can see, these are either flexions of the same term (plural vs singular) or very close terms (*zonula* meaning 'small zone' vs *zona*). Variations are explained by linguistic functions or properties. Therefore, quite a fair amount of prediction errors could be caused by linguistic variations on the same basic lexical item.

On the other hand, some equivalent definitions cannot be fathomed without the help of a domain expert. If *terminal* and *termini* could be seen as Latin flexions in the following example: `carboxy terminal`, `carboxy termini`, or in the pair `COOH-terminal`, `COOH-termini`, or in `C02H-terminal`, `CO2H termini`, the idea that *COOH*, *CO2H* and *carboxy* are equivalent forms (which makes all these pairs totally equivalent to each other) is not automatically deductible and needs expertise.

7.4 Evaluation of $IAcos$

The cosine measure has been applied, as a similarity metric in the document vector space (all the words except stopwords). The vector components are figures, representing word frequencies in the documents where the acronym has to be defined. The context of the candidate-definition is based on its close words (window of 20 to 30 words). This context is extracted from the first 10 pages returned by the Exalead search engine (this kind of context gives approximatively the same amount of words as provided by the documents). We use Exalead because specialized search engines were used to build test corpus. Then the cosine between document vector and the context of candidate definitions vector, is calculated, to predict the relevant expansion. The results presented in Table 5 show good results given by this method, i.e., 146 relevant definitions are predicted on the 204 documents (71.5% relevant definitions are ranked at the first position). The average value of the correctly predicted definitions cosine is 0.51.

Then, when shifting to the hybrid approach to improve accuracy, we calculate the $IAcos$ measure, based on the cosine and $IADef$ measures (see section 6). We select $IADef^{And}_{Dice}$ because it offers the best results in the experiments presented in section 7.2. The $IAcos$ measure consists in selecting the definitions that have the best cosine, and that are ranked at the first positions by applying $IADef$ measures.

The results are presented in Table 5 according the precision (formula (20)) and recall (formula (21)).

$$P = \frac{\text{Number of returned relevant definitions}}{\text{Number of returned definitions}} \quad (20)$$

$$R = \frac{\text{Number of returned relevant definitions}}{\text{Number of relevant definitions}} \quad (21)$$

Measures	Rate P	P (%)	Rate R	R (%)
$cosine$	146/204	71.5	146/204	**71.5**
$IADef^{And}_{Dice}$	111/204	54.4	111/204	54.4
$IAcos_1$	85/99	**85.8**	85/204	41.7
$IAcos_2$	113/142	79.5	113/204	55.4
$IAcos_3$	125/142	75.3	125/204	61.3

Table 5: Precision and Recall of the $cosine$, $IADef$, and $IAcos$ approaches ($IAcos_i$ where i represents the number of i first definitions taken into account by $IADef$).

Table 5 shows that we obtain either a best precision ($IAcos$) or a best recall ($cosine$), but not both with the same measure. This means $IAcos$ selects fewer definitions but these are more relevant. Depending on the task, the expert might want to retrieve an expansion requiring either high precision or high recall, we can use the appropriate method, i.e, $cosine$ or $IAcos$. Note that all $IAcos$ variants are on the Pareto front (11), so they are relevant. Only the $IADef$ measure used alone is dominated (see Figure 1), this is the reason why we have proposed to combine it with $cosine$ technique based on a largest context.

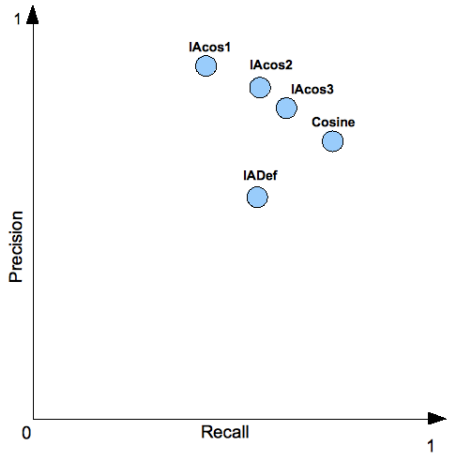

Figure 1: Pareto front of the $cosine$, $IADef$, and $IAcos$ approaches.

8 Conclusion and future work

Acronyms are widely used words that act as proper names for organizations or associations, or as shortcuts in denominating very frequent concepts or notions. As such, they are representative of the named entities issue currently tackled by the text mining scientific community. Acronyms recognition is one part of the issue, but ambiguous acronyms expansion, especially when the acronym definition is not present in the considered document, is another. This paper offers a set of quality measures to determine the choice of the best expansion for an acronym not defined in the

Web page that uses it, the $AcroDef$ and $IADef$ measures. The method uses statistics computed on Web pages to determine the appropriate definition. Measures are deeply context-based and rely on the assumption that the most frequent words in the page are related semantically or lexically to the acronym expansion. An evaluation on specialized corpora extracted from biomedical databases showed that measures still significantly operated, although contexts were much similar, and expansions very close to each other, reducing the measures ability to discriminate. However, within a context of one word (the only one with which search engines were able to retrieve pages for specific domains), the relevant definitions appeared in the first three elected by the $IADef$ measure based on Dice's coefficient with a probability of 85%. The hybrid approach presented in this paper, i.e. $IAcos$, combines a vector representation of the context (a very rich context) and $IADef$ mesure. This method improves the basic measure results precision.

$IADef$ errors are explained by the fact that they originate from too general words within contexts. If the most frequent words in the page are highly polysemous, too widely used, or vague, this has an impact on the best expansion choice, since the semantic constraint is looser. If the corpus in which acronyms have to be expanded belongs to a given domain, an interesting perspective would be to use as heuristics domain-based features (proper names, terms), or even better, a domain ontology. The experiments conducted on the biomedical corpus has clearly aimed at this direction.

Every method has its limitations and needs to be enhanced. Our approach has difficulties in building a context when the Web page in which the acronym has been found only contains a short text (a few lines for instance). Context extraction relies on words frequency as a cornerstone for thematic detection. If words are few, frequency becomes meaningless. An interesting perspective would be to represent documents as semantic vectors defined to get a thematic information on the text. These vectors project the document on a Roget-based ontology and thus do not need quantities of words to sketch a thematic environment for the acronym. That complementary information, associated with $AcroDef$ and $IADef$, would help predicting acronym definitions in the case of short texts. This work is currently undergoing as a sequel to the acronym expansion issue that we have been dealing with for a couple of years.

References

[1] E. Brill. Some advances in transformation-based part of speech tagging. In *AAAI, Vol. 1*, pages 722–727, 1994.

[2] J. Chang, H. Schtze, and R. Altman. Creating an online dictionary of abbreviations from medline. *Journal of the American Medical Informatics Association*, 9:612–620, 2002.

[3] K.W. Church and P. Hanks. Word association norms, mutual information, and lexicography. In *Computational Linguistics*, volume 16, pages 22–29, 1990.

[4] B. Daille. Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. In *The Balancing Act: Combining Symbolic and Statistical Approaches to Language, MIT Press*, pages 49–66, 1996.

[5] D. Downey, M. Broadhead, and O. Etzioni. Locating complex named entities in web text. In *Proceedings of IJCAI'07*, pages 2733–2739, 2007.

[6] C. Ferri, P. Flach, and J. Hernandez-Orallo. Learning decision trees using the area under the ROC curve. In *Proceedings of 9th International Conference on Machine Learning, ICML'02*, pages 139–146, 2002.

[7] K. Frantzi, S. Ananiadou, and H. Mima. Automatic recognition of multi-word terms: the C-value/NC-value method. *International Journal on Digital Libraries*, 3(2):115–130, 2000.

[8] F. Guillet and H.J. Hamilton. *Quality Measures in Data Mining*. Springer Verlag, 2007.

[9] M. Joshi, S. Pakhomov, T. Pedersen, and C. G. Chute. A comparative study of supervised learning as applied to acronym expansion in clinical reports. In *Proceedings of the Annual Symposium of the American Medical Informatics Association*, pages 399–403, 2006.

[10] L.S. Larkey, P. Ogilvie, M.A. Price, and B. Tamilio. Acrophile: An automated acronym extractor and server. In *Proceedings of the Fifth ACM International Conference on Digital Libraries*, pages 205–214, 2000.

[11] H. A. Leiva, S. C. Esquivel, and R. H. Gallard. Multiplicity and local search in evolutionary algorithms to build the pareto front. In *SCCC*, pages 7–13, 2000.

[12] H. Liu, A.R. Aronson, and C. Friedman. A study of abbreviations in medline abstracts. In *Proceedings of the Annual Symposium of the American Medical Informatics Association*, pages 464–468, 2002.

[13] R. Navigli. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2), 2009.

[14] G. Nenadic, I. Spasic, and S. Ananiadou. Terminology-Driven Mining of Biomedical Literature. *Bioinformatics*, 19(8):938–943, 2003.

[15] N. Okazaki and S. Ananiadou. Building an abbreviation dictionary using a term recognition approach. *22*, Bioinformatics(24):3089–3095, 2006.

[16] S. Pakhomov, T. Pedersen, and C. G. Chute. Abbreviation and acronym disambiguation in clinical discourse. In *Proceedings of the Annual Symposium of*

the American Medical Informatics Association, pages 589–593, 2005.

[17] S. Petrovic, J. Snajder, B. Dalbelo-Basic, and M. Kolar. Comparison of collocation extraction measures for document indexing. In *Proc of Information Technology Interfaces (ITI)*, pages 451– 456, 2006.

[18] V. Prince and M. Roche, editors. *Information Retrieval in Biomedicine: Natural Language Processing for Knowledge Integration*. Medical Information Science Reference, IGI Gobal, 460 pages, 2009.

[19] M. Roche and Y. Kodratoff. Pruning Terminology Extracted from a Specialized Corpus for CV Ontology Acquisition. In *Proceedings of onToContent Workshop - OTM'06, Springer Verlag, LNCS*, pages 1107–1116, 2006.

[20] M. Roche and V. Prince. Managing the Acronym/Expansion Identification Process for Text-Mining Applications. *International Journal of Software and Informatics*, 2(2):163–179, 2008.

[21] F. Smadja, K. R. McKeown, and V. Hatzivassiloglou. Translating collocations for bilingual lexicons: A statistical approach. *Computational Linguistics*, 22(1):1–38, 1996.

[22] M. Stevenson, Y. Guo, A. Alamri, and R. Gaizauskas. Disambiguation of biomedical abbreviations. In *Proceedings of the BioNLP 2009 Workshop*, pages 71–79, Boulder, Colorado, June 2009. Association for Computational Linguistics.

[23] A. Thanopoulos, N. Fakotakis, and G. Kokkianakis. Comparative Evaluation of Collocation Extraction Metrics. In *Proceedings of LREC'02*, pages 620–625, 2002.

[24] M. Torii, Z.Z. Hu, M. Song, C.H. Wu, and H. Liu. A comparison study on algorithms of detecting long forms for short forms in biomedical text. *BMC Bioinformatics*, 2007.

[25] P.D. Turney. Mining the Web for synonyms: PMI–IR versus LSA on TOEFL. *Proceedings of the 12th European Conference on Machine Learning (ECML), LNCS*, 2167:491–502, 2001.

[26] J. Vivaldi, L. Màrquez, and H. Rodríguez. Improving term extraction by system combination using boosting. In *Proceedings of the 12th European Conference on Machine Learning (ECML)*, pages 515–526, 2001.

[27] J. Xu and Y. Huang. Using svm to extract acronyms from text. *Soft Comput.*, 11(4):369–373, 2007.

[28] S. Yeates. Automatic extraction of acronyms from text. In *New Zealand Computer Science Research Students' Conference*, pages 117–124, 1999.