Disambiguation of Medical Abbreviations in French with Supervised Methods

Anaïs KOPTIENT and Natalia GRABAR
CNRS, Univ Lille, UMR 8163 – STL, F-59000 Lille, France

Abstract. Abbreviations are very frequent in medical and health documents but they convey opaque semantics. The association with their expanded forms, like *Chronic obstructive pulmonary disease* for *COPD*, may help their understanding. Yet, several abbreviations are ambiguous and have expanded forms possible. We propose to disambiguate the abbreviations in order to associate them with the proper expansion for a given context. We treat the problem through supervised categorization. We create reference data and test several algorithms. The descriptors are collected from lexical and syntactic contexts of abbreviations. The training is done on sentences containing expanded forms of abbreviations. The test is done on corpus built manually, in which the meaning of abbreviations is defined according to their contexts. Our approach shows up to 0.895 F-measure on training data and 0.773 on test data.

Keywords. Word sense disambiguation, Medical domain, Abbreviations, France

1. Introduction

Abbreviations are extremely frequent in medical and health documents, like *WHO*, *DNA*, *HT*, or *COPD*. Yet, their understanding may be complicated for patients. Indeed, the semantics of abbreviations is opaque for people without knowledge on their meaning. In such cases, it is necessary to associate the abbreviations with their extended forms in order to better understand them: *WHO - World Health Organization*, *COPD – Chronic obstructive pulmonary disease*, *HT - hypertension*. There is an important existing work on extraction of extended forms of abbreviations. Among the most frequent strategies we can mention exploitation of parentheses and of triggers like *or, meaning* [1,2], manual or automatic creation of patterns and rules [3,4,5], and exploitation of syntactic analysis [6,7]. Yet, the difficulty is that some abbreviations may have several extended forms possible [3,8,4]. For instance, an analysis of scientific literature shows that over 81% of abbreviations are ambiguous with an average of 16.6 meanings per abbreviation [8]. For example, *PC* may correspond to *personal computer*, *primary care*, *principal component*, *prostate cancer*, etc. We can mention some existing methods for disambiguating the abbreviations:

- One unsupervised method uses parallel English-German corpus [9], in which the disambiguation is done by searching the right meaning of abbreviations at cross-lingual level. On the English corpus, precision is 81% and recall 18%; on the German corpus, precision is 66% and recall 22%;
- Another unsupervised method exploits the use of collocations on the same corpus [9]. On the English corpus, precision is 79% and recall 3%; on the German corpus, precision is 82% and recall 1 %;

- One supervised method exploits both the Concept Unique Identifiers (CUIs) from the Unified Medical Language System (UMLS) and supervised learning [10]. It shows between 75 and 99% accuracy;
- Another supervised method also exploits bigrams, CUIs from the UMLS, MeSH (Medical Subjects Headings) and supervised learning algorithms [11]. This method shows between 95 and 99% accuracy.

Our purpose is to disambiguate medical abbreviations in order to choose the right extended form to be used for their explanation in a given context. We work on French texts. In what follows, we first present the methods we use to disambiguate abbreviations. We then present the results obtained and discuss them. Finally, we conclude with some issues for future work.

2. Methods

We exploit a supervised method and use classifiers implemented in the Python library Scikit-Learn [12]. In this section, we present how we collect medical abbreviations and build the reference data, and how we fix the system and evaluate the results.

Collection of abbreviations. Abbreviations are collected within the CLEAR corpus [13]. Among the 1,638 abbreviations found, 138 appear to be ambiguous. 34 ambiguous abbreviations (like *ADG – antidépresseurs de deuxième génération (second generation antidepressants)*) cannot be exploited because they show few occurrences:

- 11 abbreviations have only one occurrence of one of the extended forms,
- 7 abbreviations have only two occurrences of one of the extended forms,
- 16 abbreviations have 3 to 5 occurrences of one of the extended forms.

Hence, we work with 104 abbreviations, each of which is associated with at least 6 occurrences of its known extended forms. Table 1 shows the number of extended forms for these 104 abbreviations. We can see that abbreviations have mostly two extended forms, yet some of them may have more (up to seven for *AI*).

Table 1. Ambiguous abbreviations: number of extended forms.

	Number
2 extended forms	72
3 extended forms	18
4 extended forms	10
> 4 extended forms	4

Reference data. Documents from the corpus are first annotated by the Cordial parser [14]. For instance, *inhibitors* is lemmatized as *inhibitor* and is tagged as *Noun*. Sentences with ambiguous abbreviations are kept for building two reference sets:

- the *training corpus* is composed of 174,099 sentences which contain the extended forms of 104 abbreviations. This dataset is created automatically because the extended forms are not ambiguous;
- the *test corpus* is composed of 1,665 sentences with ambiguous abbreviations (94 out of the 104 abbreviations). This dataset is built manually: for each sentence, the right extended form of the abbreviation is defined according to its context.

Both corpora come from the CLEAR corpus. However, there is no intersection between the two datasets.

The descriptors are collected within sentences and correspond to contextual information around the abbreviations. Thus, within a five-word-window to the left and to the right starting from the abbreviations, we collect lemmas and part-of-speech tags of the context words and their position in the context. Hence, each context position is ordered. If the descriptor is present at a given position within the window its value is 1, otherwise it is 0.

Automatic disambiguation of abbreviations. We use several supervised algorithms from the Python library Scikit-Learn [12]:

- SVM Linear and SVM RBF [15] are supervised learning algorithms which can be used for classification and regression. They search a hyperplan to obtain a better division of the class parameters. We use two kernels: linear and gaussian (RBF);
- Decision Tree [16] is represented as a tree, where each choice corresponds to a given junction. The categorization is reached depending on the choices made at each step of the tree;
- MultiLayer Perceptron [17] is composed of several layers of information;
- Random Forest [18] works thanks to learning done by different decision trees trained on a subset of data.

The set of descriptors defined is exploited with each algorithms for prediction of the meaning of ambiguous abbreviations according to the context in which they occur.

Evaluation. During the training, we perform a ten-fold cross-validation. Depending on the number of meanings of abbreviations, we face two-class or multi-class categorization. Models built on the training corpus are then tested on the test corpus. When applied to the test corpus, only the first prediction for each abbreviation, which receives the highest probability, is kept. We compute standard evaluation metrics [19]: Precision, Recall and F-measure. We also compute the average values of these metrics for each algorithm. Our baseline corresponds to the categorization of meanings in the most frequent category.

3. Results

Table 2 shows the results obtained with a ten-fold cross-validation on the training corpus. These results are rather high: MultiLayer Perceptron shows up to 0.895 F-measure and Decision Tree up to 0.888. The last column of Table 2 indicates the baseline scores. We can see that the algorithms outperform the baseline at this step.

Table 2. Results obtained on the training corpus by each algorithm in 10-fold cross-validation.

Measure	SVM Linear	SVM RBF	Decision Tree	MLP	Random Forest	Baseline
Recall	0.885	0.878	0.897	0.905	0.887	0.822
Precision	0.880	0.849	0.887	0.892	0.871	0.822
F-measure	0.887	0.856	0.888	0.895	0.873	0.822

Table 3 shows the results obtained for the abbreviation disambiguation in the test corpus. We can see that now Decision Tree shows the highest results, with 0.773 F-

measure. Others algorithms show lower performance with F-measure under 0.6. On the test corpus, the baseline provides the best results. The baseline results are stable in the two corpora (train and test). We assume this is due to the fact that the proportion of the most frequent meaning of abbreviations is stable across the corpora.

Table 3. Disambiguation results obtained on the test corpus.

Measure	SVM Linear	SVM RBF	Decision Tree	MLP	Random Forest	Baseline
Recall	0.402	0.398	0.788	0.424	0.402	0.822
Precision	0.797	0.755	0.759	0.763	0.728	0.822
F-measure	0.534	0.524	0.773	0.545	0.518	0.822

Table 4 shows the total number of correctly disambiguated abbreviations in the test corpus by each algorithm. We can see that Decision Tree processed correctly the highest number of occurrences (547 out of 1,665).

Table 4. Total number of occurrences correctly classified in the test corpus.

SVM Linear	SVM RBF	Decision Tree	MLP	Random Forest	Baseline
441	516	547	523	492	0.882

Among the abbreviations that have been classified with 100% of correct predictions, we have for instance *DIU* meaning *diplôme inter universitaire (inter university diploma)* or *dispositif intra utérin (intrauterine device)* and *GH* meaning *groupe hospitalier (hospital group)* or *growth hormone*. Their correct disambiguation is mainly due to two reasons: (1) their semantics is very distinctive, and (2) each meaning has a lot of examples in the training set. 18 more abbreviations are in the same situation. Hence, their disambiguation can be obtained rather easily. Several abbreviations are categorized with different performances according to the algorithms, with F-measure going from 0 to 100%. Finally, several other abbreviations, such as *APS (amblyopie par privation de stimulus (amblyopia by stimulus deprivation)*, *antigène prostatique spécifique (specific prostatic antigen)*), *ASA (acide aminosalicylique (aminosalicylic acid)*, *American Society of Anesthesiologists*) and *HE* (*hémorrodectomie (hemorrhoidectomy)*, *encéphalopathie hépatique (hepatic encephalopathy)*), show poor results. The main reason is that the number of examples per meaning is too low and the method cannot make the decision efficiently. By comparison with the existing work, our results on training set are competitive, while the results on test set remain comparable, even if the performance decreases. Decision Tree seems to be the most appropriate for the disambiguation of abbreviations. Our results indicate nevertheless the current limitations of our work: poor training data for some abbreviations, necessity to enrich the training corpus with more examples, and to use other descriptors.

4. Conclusion and Discussion

We presented our work on disambiguation of medical abbreviations in French. We propose a method based on supervised categorization. The training is done on sentences containing extended forms, therefore semantically non-ambiguous, of

ambiguous abbreviations. The test is done on a manually built corpus, in which the correct meaning of abbreviations is defined according to the context. The results are evaluated in two ways: 10-fold cross-validation on training corpus and evaluation of models on the test corpus. Results obtained on training corpus are higher than those obtained on test corpus. For instance, Decision Tree shows an average F-measure 0.888 during training and 0.773 during test. The results of the baseline, where the meaning is assigned to the most frequent category, are higher in the test corpus. The current limits of our work is the unbalance within the training dataset, in which some meanings are poorly exemplified. This should be fixed in order to have a better balanced corpus and to obtain better results. Hence, the main issue for future work is to enrich the dataset with more examples, which should improve the processing of some abbreviations and increase overall results. We also plan to use other descriptors, like BERT.

Acknowledgments. This work was partly funded by the French National Agency for Research (ANR) as part of the *CLEAR* project (*Communication, Literacy, Education, Accessibility, Readability)*, ANR-17-CE19-0016-01.

References

[1] Park Y and Byrd RJ. Hybrid text mining for finding abbreviations and their definitions. In *Empirical Methods of Natural Language Processing*, pages 126–33, 2001.
[2] Schwartz A S and Hearst M A. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Pacific Symposium on Biocomputing*, pages 451–456, 2003.
[3] Chang JT, Schtze H, and Altman RB. Creating an online dictionary of abbreviations from MEDLINE. *J Am Med Inform Assoc*, 9(6):612–20, 2002.
[4] Liu H and Friedman C. Mining terminological knowledge in large biomedical corpora. In *Pac Symp Biocomput,* pages 415–26, 2003.
[5] Ao H and Takagi T. ALICE: An algorithm to extract abbreviations from MEDLINE. *J Am Med Inform Assoc,* 12(5):576–586, 2005.
[6] Pustejovsky J, Castano J, Cochran B, Kotecki M, and Morrell M. Automatic extraction of acronym-meaning pairs from Medline databases. In *MEDINFO*, pages 371–5, 2001.
[7] Yoshida M, Fukuda K, and Takagi T. PNAD-CSS: a workbench for constructing a protein name abbreviation dictionary. *Bioinformatics*, 16:169–75, 2000.
[8] Liu H, Aronson AR, and Friedman C. A study of abbreviations in Medline abstracts. In *Ann Symp Am Med Inform Assoc (AMIA),* pages 464–8, 2002.
[9] Widdows D, Peters S, Cederberg S, Chan CK, Steffen D, and Buitelaar P. Unsupervised monolingual and bilingual word-sense disambiguation of medical documents using umls. In *BioNLP*, pages 1–8, 2003.
[10] Liu H, Lussier Y, and Friedman C. A study of abbreviations in the UMLS. In *Ann Symp Am Med Inform Assoc (AMIA),* Washington, 2001.
[11] Stevenson M, Agirre E, and Soroa A. Exploiting domain information for word sense disambiguation of medical documents. *J Am Med Inform Assoc*, 19:235–240, 2012.
[12] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, and Duchesnay E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
[13] Grabar N and Cardon R. Clear – simple corpus for medical French. In *Workshop on Automatic Text Adaption (ATA)*, pages 1–11, 2018.
[14] Laurent D, Nègre S, and Ségula P. L'analyseur syntaxique Cordial dans Passage. In *Traitement Automa-tique des Langues Naturelles (TALN)*, 2009.
[15] Platt JC. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1998.
[16] Quinlan JR. *C4.5 Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.
[17] Rosenblatt F. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. *Spartan Books,* Washington DC, 1961.
[18] Breiman L. Random forests. *Machine Learning*, 45(1):5–32, 2001.
[19] Sebastiani F. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.