A Supervised Abbreviation Resolution System for Medical Text

Pierre Zweigenbaum[1], Louise Deléger[1], Thomas Lavergne[1],
Aurélie Névéol[1], and Andreea Bodnari[1,2]

[1] LIMSI-CNRS, rue John von Neumann, F-91400 Orsay, France
[2] MIT, CSAIL, Cambridge, Massachusetts, USA

Abstract. We present our participation in Task 2 of the 2013 CLEF-eHEALTH Challenge, whose goal was to determine the UMLS concept unique identifier (CUI), if available, of an abbreviation or acronym. We hypothesize that considering only the abbreviations of the training corpus could be sufficient to provide a strong baseline for this task. We therefore test how a fully supervised approach, which predicts the CUI of an abbreviation based only on the abbreviations and CUIs seen in the training corpus, can fare on this task. We adapt to this task the processing pipeline we developed for CLEF-eHEALTH Task 1, entity detection: a supervised MaxEnt model based on a set of features including UMLS Concept Unique Identifiers, complemented here with a rule-based component for document headers. This system confirmed our hypothesis, and was evaluated at 0.664 accuracy (strict) and 0.672 accuracy (relaxed), ranking second out of five teams.

Keywords: Abbreviation normalization, Natural Language Processing, Medical records, Machine learning

1 Introduction

The 2013 CLEF-eHEALTH challenge [9] aims to develop methods and resources that make electronic medical records (EMRs) more understandable by both patients and health professionals. The challenge spans three tasks. The second task focuses on the resolution of abbreviations into UMLS concept unique identifiers (CUIs). We present here our participation in the second task and develop a system that can determine the CUI of an abbreviation. We propose a solution that combines a simple feature set with external knowledge gathered from the UMLS Metathesaurus.

Clinical notes often use abbreviations and acronyms (henceforth collectively named 'abbreviations' in this paper). While the number and variety of abbreviations is unlimited thanks to the plasticity of language and the inventiveness of the human mind, the distribution of abbreviations can be expected to follow a Zipfian law, with a small number of distinct abbreviations accounting for a large part of the total number of occurrences of abbreviations. We therefore hypothesize that the sample of abbreviations present in the training corpus provided by CLEF-eHEALTH for the task should contain most frequently occurring abbreviations and should hence account for a large part of the occurrences of abbreviations in the test corpus.

We present a supervised abbreviation resolution system specifically adapted to the abbreviations of the CLEF eHealth corpus. Our system learns a MaxEnt model from the training data, based on a simple feature set that combines UMLS knowledge with information gathered from the EMR text. We present the system design, its results on the 2013 CLEF-eHEALTH [9] test data, and discuss its merits and limitations.

2 Related work

Abbreviations and acronyms are pervasive in technical and scientific text, including in health and life sciences. They create obstacles to various information extraction tasks, for instance for coreference resolution in electronic medical records [10]. This has motivated work on abbreviation detection and expansion on various types of texts, the most prevalent of which for health are the scientific literature [6, 8, 1] and patient records [11, 3]. Initial work aimed to collect acronyms and their expansions from large corpora [6, 8], in expressions such as *the numbers of underrepresented minorities (URMs)*[1] where the expansion of an abbreviation is directly provided in the input text. This is however not always the case, and is even fairly rare in patient records.

When abbreviation expansions are not explicit in source texts, abbreviation processing can be divided into two steps. The first is the detection of abbreviations, which determines which expressions in a text are abbreviations (or acronyms) [11]. This can be compared to an entity detection task. The second step is the resolution of abbreviations: given an expression, which may map to multiple expansions, select the appropriate expansion in context. If all possible expansions are available, this is similar to a disambiguation task [3]. Kim et al. [3] disambiguate abbreviations with a multi-class SVM classifier trained on feature vectors including the five preceding and following words (occurring at least five times in the corpus). They train on a corpus of 9,963 clinical notes in which full forms have automatically been replaced with abbreviations, following Pakhomov [5], using the inventory of abbreviations in the UMLS Specialist Lexicon file LRABR. They test their method on a corpus of 37 hand-annotated clinical notes and achieve an F-measure of 0.660 (Precision = 0.683, Recall = 0.637) in exact match and 0.754 in partial match.

Task 2 of the ShARe CLEF eHealth 2013 Challenge addresses the abbreviation disambiguation step. Gold standard annotations of abbreviation boundaries are given for the abbreviations present in clinical notes, which removes the need to address the abbreviation detection step: the task consists of disambiguating and normalizing each abbreviation into a UMLS Concept Unique Identifier

[1] Example from the MEDSTRACT corpus: `http://medstract.org/gold-standards.html`.

(CUI), or the *CUI-less* label if no UMLS CUI is available. It is close to the task defined in [3], but it is to our knowledge the first time a dataset of this size is provided to address abbreviation disambiguation in clinical notes into a reference terminology. Besides, a rapid examination of the abbreviations found in the training dataset shows that a number of them are not present in the Specialist Lexicon LRABR file: for instance, *oxygen saturation* abbreviated as *O2 sat* or *02 sat* (this latter example uses a zero instead of the letter 'O') is not listed among the 4 abbreviations of *oxygen saturation* in LRABR (*O2 saturation, O2sat, SO2, SpO2*), so the semi-supervised method of [5, 3] might not apply here.

3 Materials and methods

3.1 Data

The corpus used for the 2013 CLEF-eHEALTH challenge consists of de-identified plain text EMRs from the MIMIC II database, version 2.5 [7]. The EMR documents were extracted from the intensive-care unit setting and included discharge summaries, electrocardiography reports, echography reports, and radiology reports. The training set contained 200 documents and a total of 94,243 words, while the test set contained 100 documents and a total of 87,799 words (see Table 1).

Abbreviations and acronyms are annotated in terms of span and UMLS Concept Unique Identifier (CUI) when available. If no CUI is available for the abbreviation the *CUI-less* code is used. The training set contained 3, 805 annotations, while the test set contained 3, 774 annotations (see Table 1). CUI-less abbreviations accounted for about 5% in the training set and 6% in the test set.

Table 1. Description of training and test data sets. Distinct CUIs do not include the *CUI-less* code.

	Training	Test
Documents	200	100
Words	94,243	87,799
Abbreviations	3,805	3,774
Distinct CUIs	696	706
CUI-less abbreviations	181	221

3.2 System design

Document headers (see details below) contain pre-formatted fields, some of which contain abbreviations. Since these are very regular fields, their CUIs are not ambiguous, and simple rules can handle them. We therefore divided the problem into two parts:

- Rule-based resolution of abbreviations in document headers (Section 3.4);
- Supervised resolution of abbreviations in document body (Section 3.3).

3.3 Supervised resolution of abbreviations in document body

We used a supervised linear-chain Conditional Random Fields (CRF) system which we restricted to Maximum Entropy mode (see below). We trained a model using 10-fold cross validation on the training set data, keeping 1/11th of the data for final tuning of the model. We first present how we formulate the problem. We then describe the pre-processing steps we performed on the datasets and the model feature set together with the CRF feature patterns.

Problem formulation Abbreviation resolution is defined here as the determination of the CUI of an abbreviation, or else the attribution of the *CUI-less* label. An abbreviation may be ambiguous, i.e., be found with different CUIs in different contexts. For instance, in the training set, *CATH* is used for *Catheterization* [C0007430] and for *Drainage Catheters* [C0879005]. When ignoring case, 526 distinct abbreviations are found in the training set, 94 of which are ambiguous given the entries in the training set.

When all candidate CUIs for an abbreviation are known, abbreviation expansion can be addressed as a supervised classification task—this is what we do here. We label each abbreviation of the training set with its gold standard CUI, using a B-I-O scheme where each CUI introduces a B- and possibly some I- labels. For instance, the two tokens of abbreviation “O2 sat” are labelled as B-C0523807 and I-C0523807. This results in 738 distinct labels. An obvious limitation of this design is that only those CUIs seen in the training set can be assigned in the test set, but we found it worth trying.

We used the Wapiti [4][2] implementation of CRFs because it is fast and offers convenient patterns (e.g., patterns using regular expressions on field values).

Data pre-processing Our data pre-processing and feature production architecture is schematized in Figure 1. Before using the challenge corpora for training and testing, we performed the following pre-processing steps:

- the training and test corpora provided by the challenge organizers were de-identified and thus contained special de-identification marks; to turn de-identification code into more normal phrases, we performed re-identification with pseudonyms on the input text.
- EMR documents present in general a well-structured form, with a header, document body, and a footer. The header and footer contain information relevant to clinical administration, but the disorder NPs are only encountered inside the document body. We thus removed the header and footer from the EMR documents and performed analysis on the document body only. Since headers may contain abbreviations, we handled them specifically (see below).

[2] `http://wapiti.limsi.fr/`

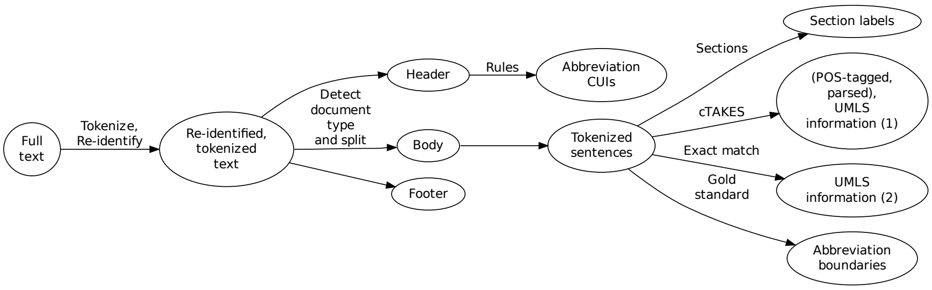

Fig. 1. Diagram of feature production.

System features Given a sentence $s = \ldots w_{-2}w_{-1}w_0w_1w_2\ldots$ and a token of interest w_k, we define features over w_k and n-grams centered at w_k.

1. **Lexical and morphological features**: we include information on the token in the form of unigrams over w_{k-1}, w_k, w_{k+1} and bigrams over $w_{k-1}w_k$. Additionally we include as unigram features over w_k token prefixes ranging from 1 to 4 characters. We also add a 5-gram feature which detects patterns containing two slashes, such as "m/r/g", which may help disambiguate between disorders and abbreviations, and apply it over w_{k-4}, w_{k-2}, w_k (the non-slash positions of the pattern).
2. **Document structure features**: the feature set contains as features the document type (e.g., radiology report, discharge summary) and the section type (e.g., Findings, Laboratory data, Social History). We extract the section type using a rule-based section extraction tool that identifies the occurrence of section names within the EMR. The section extraction tool uses a list of 58 manually defined section names. Both document type and section type are unigram features over w_k.
3. **UMLS features**: we include UMLS information from two sources. We first run cTAKES over the texts and keep concept unique identifiers (CUIs, defined over unigrams w_k). We use an additional UMLS mapping where we directly search for UMLS strings within the EMR text through exact match and include the concept unique identifier (CUI) of the identified phrase. The direct UMLS mapping features are unigram features over w_k.
4. **Abbreviation feature**: gold standard boundaries for abbreviations are provided. These are used as unigram features over w_k.

All features pertaining to multi-token expressions instead of only single tokens (for instance, being a UMLS term with a given CUI, or being an abbreviation) are encoded with the Begin Inside Outside (B-I-O) scheme: given a label L, the first token is labeled B-L, the next tokens are labeled I-L, and tokens not having this feature are labeled O. An advantage of linear-chain CRFs is that they can include label bigrams in the model they learn. However, in our experiments on the training set, we were not able to use label bigrams because of the large number of labels (738 in the model we finally used): with label bigrams, the number of generated features for a CRF pattern which involves x observed values and y labels is $x \times y^2$, and it seems that the feature space generated when we tested with these label bigrams was to vast for the CRF to find a solution when training. Having no label bigrams means that our CRF is used basically as a Maximum Entropy classifier, with no dependency on the previous label. The total number of features generated for evaluation by the CRF for this model is about 50 million, among which it selects about 400,000.

3.4 Rule-based resolution of abbreviations in document headers

We examined abbreviations in the training set headers and wrote rules to label them. These rules are shown in Table 2. They handle the 6 abbreviations found in the document headers of the training set. The resulting annotations are merged with those obtained by the CRF.

Table 2. Task 2. Rules to address abbreviations in document headers.

Document type	Pattern	Abbrev = CUI
DISCHARGE	Sex : M	M = C0024554
DISCHARGE	Sex : F	F = C0015780
ECG	ECG	ECG = C0013798
ECHO	ECHO	ECHO = C0013516
RADIO	PORTABLE AP	AP = C1999039
RADIO	CT HEAD	CT = C0040405

4 Results and discussion

4.1 Evaluation metrics

We evaluate the system's ability to correctly generate the codes (CUIs or *CUI-less*) of abbreviations. The evaluation measure is the accuracy of codes, defined as

$$Accuracy = \frac{Correct}{Total} \tag{1}$$

where
Correct= Number of pre-annotated acronyms/abbreviations with correctly generated code;
Total= Number of pre-annotated acronyms/abbreviations.

In some cases the human annotators generated a ranked list of codes for a given abbreviation. This makes it possible to define two variants of the accuracy score. In the strict variant, only the top code selected by the annotators (one best) is considered. In the relaxed variant, a code is considered correct if it is contained in the full list generated by the annotators (n-best).

4.2 Rule-based resolution of abbreviations in headers

Rule-based resolution of abbreviations in headers obtained the results displayed on Table 3. 109 abbreviations were located in the headers; the rules identified 100, out of which 97 matched the gold standard. The two *substitutions* (the gold standard CUI was different from that proposed by the system) are both cases where human annotators marked as *CUI-less* an *M* (Male) or *F* (Female) abbreviation in the header, which seems to be an error in the gold standard.[3] Since we did not constrain rules to apply to gold standard abbreviation boundaries, the system also identified an *M* abbreviation which the human annotators forgot to mark in a document header (*insertion*).[4] If this is correct, rule-based resolution of abbreviations had perfect accuracy.

Table 3. Headers: Rule-based resolution of abbreviations. ECG is null because there was no ECG report in the test corpus. The two substitutions and the insertion actually seem to be errors in the gold standard.

Abbreviation	CUI	GS	Correct	Substitution	Deletion	Insertion	Accuracy
M	C0024554	42	41	1		1	0.98
F	C0015780	31	30	1			0.97
ECG	C0013798	0	0				
ECHO	C0013516	12	12				1.00
AP	C1999039	12	12				1.00
CT	C0040405	2	2				1.00
Others		10	0		10		0.00
Total		109	97	2	10	1	0.89

GS = Gold Standard. Accuracy = Correct / (Correct + Substitution + Deletion).

However, the headers were processed only through these rules, and other abbreviations in the headers were thus not examined. 10 such abbreviations were missed (*deletions*), so that overall, our resolution of abbreviations in headers achieved an accuracy of 0.89 against the gold standard. All in all, the hundred of abbreviations present in headers are only a small part of the 3,774 abbreviations of the test corpus (about 3%), so this part of the method only contributed a small fraction of the performance of the system.

4.3 Resolution of abbreviations in full documents

This section presents the global results that we obtained on the full documents, merging codes obtained through rule-based and supervised methods. We submitted one run which achieved an accuracy of 0.664 under the strict evaluation and 0.672 under the relaxed setting.

[3] 01163-001840-DISCHARGE_SUMMARY.txt:235–236,
00534-017453-DISCHARGE_SUMMARY.txt:216–217

[4] 15230-012950-DISCHARGE_SUMMARY.txt:219–220

The training set contained 696 distinct CUIs (plus the *CUI-less* label), whereas the test set contained 706 distinct CUIs (plus *CUI-less*) (see Table 1). When training our final CRF model we kept 1/11th of the corpus for development, resulting in a model which has knowledge for 654 CUIs (plus *CUI-less*). Only 346 of these were present in the test set, which means that our system could only identify about one half of the 706 distinct CUIs of the test set.

However, considering the number of occurrences of these CUIs draws a quite different picture. Table 4 shows that among the 3774 occurrences of abbreviations contained in the test set, 2906 had CUIs seen in the training corpus (including 221 occurrences of *CUI-less*, see Table 5). Correctly identifying these known CUIs would have led to an accuracy of 2906/3774 = 0.770, which would outperform the best system of the challenge (0.719). Our hypothesis according to which considering only the abbreviations of the training corpus could be sufficient to provide a strong baseline is therefore confirmed.

Table 4. Full documents: Resolution of abbreviations (strict evaluation) is much better for seen CUIs than for unseen CUIs. A CUI is seen if it was fed to the CRF model during training.

CUI	GS	Correct	Substitution	Deletion	Insertion	Accuracy
Seen	2906	2354	456	96	285	0.810
Unseen	868	151	650	67		0.174
All	3774	2505	1106	163	285	0.664

GS = Gold Standard. Accuracy = Correct / (Correct + Substitution + Deletion).

Besides, our system correctly identified the CUIs of 2354 occurrences of abbreviations, i.e., 81% of the best it could have done given its inherent limitations. In the case of unseen abbreviations, our system can only be right if the abbreviation is *CUI-less* and the system does assign this code to it; this occurred for 151 occurrences. Whether or not an abbreviation had a CUI did not affect much the performance of the system, as can be seen on Table 5.

Table 5. Full documents: Resolution of abbreviations (strict evaluation) fares about as well on abbreviations with and without CUIs.

CUI	GS	Correct	Substitution	Deletion	Insertion	Accuracy
CUI-less	221	151	57	13	48	0.683
with CUI	3553	2354	1049	150	237	0.663
All	3774	2505	1106	163	285	0.664

GS = Gold Standard. Accuracy = Correct / (Correct + Substitution + Deletion).

We conclude this discussion with a word on insertions and deletions. These were not expected given the task definition: given gold standard abbreviation boundaries, determine the CUI of the abbreviation. Using a CRF with a B-I-O scheme allowed us to keep the very same framework as we prepared for Task 1 of the challenge, Entity Recognition [2]. Its drawback is that without additional constraints, it decides where to set the boundaries of the CUI or CUI-less codes that it assigns to its input tokens. When this created boundary changes, they caused a mismatch between the gold standard and system abbreviations, which can be counted as a deletion (missing gold standard abbreviation) combined to an insertion (spurious system abbreviation).

For instance, the gold standard considered the string *NCAT* as two tokens *NC* [CUI-less] *AT* [CUI-less] while the system tagged it as one token: *NCAT* [CUI-less]. Conversely, the system tagged *02 sat* [C0523807] as two tokens *02* [C0030054] *sat* [C0523807], and tagged *LE's* [C0023216] as three tokens *LE* [C0023216] *'* [C1444662] *s* [C0023901]. A split into multiple tokens occurred more frequently than the reverse situation, which explains why insertions (285) are more numerous than deletions (163). Since the gold standard boundaries are given, a post-processing step could be added in the future to predict a code for the actual abbreviation span based on the CRF output. The cleanest way to cope with this issue though will be to rewrite the system to quit the linear-chain framework of our Task 1 system and switch to a standard Maximum Entropy (or other) classifier.

5 Conclusion and perspectives

We present an abbreviation resolution system prepared for participation in Task 2 of the 2013 CLEF-eHEALTH challenge. We design our system as a derivation of our Task 1 framework into a supervised Maximum Entropy classifier with lexical, document structure, and UMLS features, complemented by rule-based processing of abbreviations in document headers. Our system obtains an accuracy of 0.664 (strict) and 0.672 (relaxed), which gives it a second position among five systems. We have tested the bold hypothesis that relying only on the CUIs seen in the training set is enough to provide a strong baseline. The data in the test set and the obtained results show that this hypothesis holds. We have seen that in principle this framework might even fare better than the current best challenge participant. Further work can go into two directions. One is to keep the current framework and strengthen it, for instance by removing deletions and insertions, and by studying the current causes of substitutions. The other is to move to dynamic generation of candidate expansions. This may require to abandon the neat supervised classification framework which was possible here, but is needed to extend processing to abbreviations unseen in the training set or in existing databases such as the UMLS Specialist Lexicon LRABR.

6 Acknowledgments

We acknowledge the Shared Annotated Resources (ShARe) project funded by the United States National Institutes of Health with grant number R01GM090187. The last author was funded by the Châteaubriand Science Fellowship 2012–2013.

This work was partly funded through project Accordys[5] funded by ANR under grant number ANR-12-CORD-0007-03.

References

1. Paolo Atzeni, Fabio Polticelli, and Daniele Toti. An automatic identification and resolution system for protein-related abbreviations in scientific papers. In C. Pizzuti, M.D. Ritchie, and M. Giacobini, editors, *EvoBIO 2011*, number 6623 in LNCS, pages 171–176. Springer-Verlag, Berlin Heidelberg, 2011.
2. Andreea Bodnari, Louise Deléger, Thomas Lavergne, Aurélie Névéol, and Pierre Zweigenbaum. A supervised named-entity extraction system for medical text. In *Proceedings of CLEF 2013*, 2013. To appear.
3. Youngjun Kim, John Hurdle, and Stphane M. Meystre. Using UMLS lexical resources to disambiguate abbreviations in clinical text. In *AMIA Annu Symp Proc*, pages 715–722, 2011.
4. Thomas Lavergne, Olivier Cappé, and François Yvon. Practical very large scale CRFs. In *ACL Proc*, pages 504–513, 2010.
5. Serguei Pakhomov. Semi-supervised maximum entropy based approach to acronym and abbreviation normalization in medical texts. In *Proc 40^{th} ACL*, pages 160–167, Philadelphia, PA, 2002. ACL.
6. James Pustejovsky, José Casta no, Brent Cochran, Maciej Kotecki, and Michael Morrell. Automatic extraction of acronym-meaning pairs from MEDLINE databases. In *MEDINFO 2001*, number 1 in Studies in health technology and informatics, pages 371–375. IOS Press, 2001.
7. M. Saeed, M. Villarroel, A.T. Reisner, G. Clifford, L. Lehman, G.B. Moody, T. Heldt, T.H. Kyaw, B.E. Moody, and R.G. Mark. Multiparameter intelligent monitoring in intensive care II (MIMIC-II): A public-access ICU database. *Clinical Care Medicine*, 39:952–960, 2011.
8. Ariel S. Schwartz and Marti A. Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text. In *Pacific Symposium on Biocomputing*, pages 451–462, Kauai, Hawaii, 2003. World Scientific Pub Co Inc.
9. Hanna Suominen, Sanna Salanterä, Wendy W. Sumitra Velupillai Chapman, Guergana Savova, Noemie Elhadad, Sameer Pradhan, Brett R. South, Danielle L. Mowery, Gareth J.F. Jones, Johannes Leveling, Liadh Kelly, Lorraine Goeuriot, David Martinez, and Guido Zuccon. Overview of the ShARe/CLEF eHealth evaluation lab 2013. In *Proceedings of CLEF 2013*, Lecture Notes in Computer Science, Berlin Heidelberg, 2013. Springer. To appear.
10. Özlem Uzuner, Andreea Bodnari, Shuying Shen, Tyler Forbush, John Pestian, and Brett South. Evaluating the state of the art in coreference resolution for electronic medical records. *Journal of American Medical Informatics Association*, 17:514–518, February 2010.
11. Yonghui Wu, S. Trent Rosenbloom, Joshua C. Denny Randolph A. Miller Subramani Mani, Dario A. Giuse, and Hua Xu. Detecting abbreviations in discharge summaries using machine learning methods. In *AMIA Annu Symp Proc*, pages 1541–1549, 2011.

[5] Accordys: *Agrégation de Contenus et de COnnaissances pour Raisonner à partir de cas de DYSmorphologie fœtale*, Content and Knowledge Aggregation for Case-based Reasoning in the field of Fetal Dysmorphology (ANR 2012-2015).