ARBOREx: Abbreviation resolution based on regular expressions for BARR2[⋆]

Fernando Sánchez-León

f.sanchez.lcmcvp@gmail.com

Abstract. ARBOREx, a system for abbreviation recognition and resolution in biomedical texts based on a curated set of lexical resources, is presented. For ShortForm-ExtendedForm (SF-EF) explicit relations found in the text a ranked dynamic regular expression approach is used, complemented by access to Google Translation Services when needed. On the other hand, for EF resolution when a SF candidate has been identified in the source text, a knowledge-based approach is adopted, implemented as a set of simplified patterns that compile to regular expressions applied over the text stream. These patterns perform EF selection exploiting linguistic form (word co-occurrence in discourse) as a means to map SF to term meaning. A very preliminary version of a conceptual classification of SF-EF pairs is used in some of the patterns. Besides, a simple bag-of-words model that computes the overlapping of the content words in the EF to those in the text is also used as a fallback strategy. The system outlined has been used in the BARR2 track, described elsewhere [1]. On the BARR2 datasets, ARBOREx achieved high precision and recall for both tasks, with an F1 score of 88.42 for task 1 and 83.17 for task 2.

Keywords: Abbreviation recognition · abbreviation resolution · biomedical texts · dynamic regular expressions · approximate string matching · regular expression sense resolution · biomedical lexical resources · Electronic Health Records.

1 Introduction

The maturity of the adoption of Electronic Health Records (EHRs) in the Spanish Social Security Health system is rapidly increasing the need for tools to make these records accessible, interoperable and, on top of that, owned by the patients. These are only some of the *ambitions* of the EU digital single market for

[⋆] This work is an offering to the memory of Pablo Sánchez Creus and Antonio León Martínez-Campos. It is also dedicated to the healthcare professionals who treated the author at Hospital Universitario HM Sanchinarro and Hospital de La Princesa, both in Madrid —Alfons Serrano Maestro, Beatriz Aguado Bueno, Ángela Figuera Álvarez, María Luisa García Buey, Jaime Pérez de Oteyza and María Luisa Bengoechea Casado—, and specially to an anonymous life-giving donor.

eHealth[1]. In this global scenario, content analysis and unambiguous understanding of these records —usually written in an informal and rather jargon language variety/ies— is urgently needed. As part of this challenge on applying natural language processing techniques to different domains in the new digital society, IberEval 2018 has launched BARR2, the Biomedical Abbreviation Recognition and Resolution 2nd edition competition.

This paper presents ARBOREx, a system developed for BARR2 track. We won't go into the extensive previous work done on medical abbreviation identification and will simply describe our system in this paper. The interested reader can refer to Dogan *et al.* [2] for an overview of algorithms, corpora and formats used for the identification of abbreviations in biomedical literature. Section 2 describes the resources we have collected and enriched for our system. Some general issues on the development of ARBOREx are presented in section 3. Section 4 explains our alignment strategy for task 1, while section 5 describes the methods used for task 2. System results are presented in section 6 along with an extensive discussion of error analysis. Finally, section 7 includes some concluding remarks.

2 Resources

This section describes the resources built from scratch (or collected, adapted and further developed) specifically for the abbreviation recognition system presented in this paper. A specialized dictionary for human consumption has been used a as seed for the construction of our lexical resources. A preliminary version of the format for these resources is also presented.

2.1 Sources and Resources

There are no good NLP systems without better language resources. This principle has guided the implementation of our solution to BARR2 track.

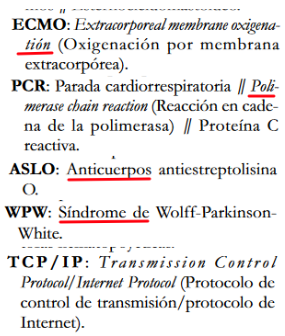
To this end, we have gathered several different sources and treated them to painstakingly construct the resources used by our system. The aim in this phase of the system development has been to **compile** rather than simply pile the resources at hand. These resources are briefly described in this section.

Datasets We have used three (bio)medical datasets for resource building: a web dataset constructed mining several websites with biomedical information (scielo, Elsevier, MedlinePlus), a collection of 80 medicine introductory textbooks from various subdisciplines and BARR2 background dataset [3]. These sets have been (partially) used with preliminary versions of our system in order to gather SF-EF pairs to enrich our core resources (either providing new SF-EF pairs, or,

[1] https://ec.europa.eu/health/sites/health/files/ehealth/docs/2018_ehealth_infographic_en.pdf (last consulted 07/13/2018).

crucially, pointing to new EF variants for known SFs)[2]. Besides, they have been used for `grep`ing tasks (no indexation of the datasets has been performed), as an aid to model the *cotext* of a SF occurrence (for task 2; see section 5.1).

DSM [4] A PDF version of the medical abbreviations dictionary published by the Spanish Ministerio de Sanidad y Consumo has been found on the web. After downloading, it has been OCRed and a *script* implemented to convert to a custom XML format. Careful reading of the output has allowed us to detect and correct errata and minor errors. Figure 1 shows some of them.



ECMO: *Extracorporeal membrane oxigenatión* (Oxigenación por membrana extracorpórea).

PCR: Parada cardiorrespiratoria // *Polimerase chain reaction* (Reacción en cadena de la polimerasa) // Proteína C reactiva.

ASLO: Anticuerpos antiestreptolisina O.

WPW: Síndrome de Wolff-Parkinson-White.

TCP/IP: *Transmission Control Protocol/Internet Protocol* (Protocolo de control de transmisión/protocolo de Internet).

Fig. 1. Some of the errors (and misplaced? acronyms) found in the original *DSM*.

2.2 Resource format

Apart from units of measurement (some 500 symbols), iteratively running our prototype on the available datasets, and performing some experimentation on similarity thresholds and extraction strategies being used, we have collected nearly 15,000 SF-EF pairs, although many of them show multiple EF variants. As an example, consider the following entry taken from our lexicon:

```
 <entry abbr="CREST" type="acr" id="000947" expan="calcinosis cutánea,
fenómeno de Raynaud, disfunción esofágica, esclerodactilia y
telangiectasia, síndrome" en="calcinosis cutis, Raynaud’s phenomenon,
esophageal dysfunction, sclerodactyly and telangiectasia"
variants="calcinosis, síndrome de Raynaud, alteraciones esofágicas,
esclerodactilia y telangiectasias // calcinosis, fenómeno de Raynaud,
dismotilidad esofágica, esclerodactilia y telangiectasias //
```

[2] Even though, depending on the author and the task at hand, different names have been used for both SF —abbreviation, acronym, symbol (since chemical symbols and units of measurement have to be extracted as well), initials— and EF —definition, long form—, we use the terms ShortForm (SF) and ExpandedForm (EF).

```
calcinosis, Raynaud, hipomotilidad esofágica, esclerodactilia o
esclerodermia, telangiectasias" prefer="en" concept="C_FINDING"
src="DSM" />
```

Besides expan and en, for EF in Spanish and English, respectively, a set of variants for (Spanish) EF has been added after processing our datasets. variants' multiple values are separated by a double slash (//)[3]. As it can be seen (and will be explained in section 5.1), there exists a concept attribute holding a conceptual classification of the coded medical term[4]. The purpose of the prefer feature will be explained in section 5.

3 Development

A preliminary version of ARBOREx was developed in Python. Unfortunately, this language was soon abandoned due to its poor performance processing regular expressions (REs). Since REs are a fundamental part of any information extraction task, we turned back our attention to Perl for its huge ecosystem and fast prototyping features[5]. ARBOREx has been developed as an OO library with a clean API that is used by two scripts, one for each task in BARR2. There are also a number of other scripts to perform extraction of variant EFs to known SFs, resource updating, compilation of resources used by the library, alignment guessing…

The system is strongly based on lexical resources (even for task 1) and on linguistic heuristics that try to correlate linguistic form with meaning (especially for task 2). However, no NLP pipeline is used and no linguistic processing as such is performed on the input text, not even tokenization. This decision was motivated by the high proportion of errors in the tokenization phase produced by some of the tokenizers tested and by the fact that no linguistic analysis was going to be performed on the clinical case files. Moreover, processing the input text as a stream of bytes have allowed us to implement fast ways to deal with

[3] Alongside with English, there are EFs in other languages —at the moment, French, Catalan, Galician and, of course, Latin. Variants can also be provided for any of these languages, prefixing the attribute name with the two-letter language code (ISO 639-1) followed by an underscore character. We are aware of the extremely compact and not very elegant XML representation of lexical and terminological entries, but time constraints for the track have forced us to model information in this simplistic way. The whole work described in this paper has been developed in about 10 weeks.

[4] This constitutes a very preliminary version of such an expressive characterization. In its current state of development, it has only been applied to approximately half of the entries in ARBOREx lexical resources.

[5] There are many benchmarks on the web comparing the efficiency in RE processing for these two languages. According to them, Perl is 2 to 10 times faster for this particular feature. Perl is even faster for RE processing than Python PyPy (see, for instance, http://github.com/mariomka/regexp-benchmark). Moreover, further improvements can be obtained if *idiomatic* Perl is used. For instance, being equivalent if locale rules are in effect, \w is 25% faster than [[:word:]].

(approximations of) word lexemes and multi-word stop fragments, as will be discussed in the following sections.

Sections 4 describes our set of alignment strategies for task 1, while section 5 presents a prototype version for the more challenging task 2, where correct EF selection has to be performed on top of simple text-lexicon alignment.

4 Task 1

For task 1, participants had to extract all SF-EF pairs where EF is explicitly mentioned in the text. Both (bio)medical and general abbreviations as well as symbols had to be extracted as well. Our method is described in this section.

After spotting all SF candidates in each text, using contextual cues and a very general RE for SF character form, offset and a context window for each SF is memory cached to be later search for available EF to be aligned with. Contextual cues are also used to hypothesize the direction (left, right or both) where the EF should be found (if present)[6].

Using SF string —possibly transformed removing dots, hyphens and other non alphanumeric characters—, we compile REs dynamically and on demand. This means that first we build a RE to be matched with EF word initials (let's call it Initials) to the direction indicated by our previously matched spotting cue. First attempt to match word initials does not allow words between SF and EF candidates other than those whose initials match with the so built RE. If this matching fails, another RE is built and tried to match with the cached context. Dynamically built REs include stop words[7], multicharacter word matching and even character permutations over SF[8]. We also allow for non relevant, intertwined words in the EF candidate.

This is the common path used to align unknown SF-EF pairs. However, for known SFs (that is, SFs included in our lexicon), the system allows for a shortcut. In this situation, for each EF available in the lexicon for the known SF, the EF is matched approximately with the context. For this approximate string matching task (ASM), we have modified Perl's built-in RE engine and substituted it by the approximate RE matching library TRE[9]. A similarity threshold —that

[6] *Both* has not been implemented for BARR2 track since they are rare in clinical texts. However, this type is potentially productive in other medical texts. We refer to cases like "Analizar en los cánceres inflamatorios de mama (inflammatory breast carcinoma [IBC])" or "un segmento con flexión pasiva (PB, de sus siglas en inglés passive bending section) contiguo al segmento móvil del endoscopio". Both Spanish and English EFs can be captured in these cases.

[7] Actually, since the input is a byte stream and they are used in a RE context, we have a stop word RE set rather that a stop word list. ARBOREx also uses a similar stop word RE set for English.

[8] As an example, it is through permutations that the system is able to match "gonadotropina coriánica humana beta" with SF "b-GCH".

[9] `https://github.com/laurikari/tre/`

is converted to a cost based on EF length— is used. During development phase, this threshold has been lowered, in order to gather EF variants for lexicon update. The approximate RE matching is performed twice —for the text fragment containing the lexicon EF, and reversely for the lexicon EF containing the text fragment. The latter matching operation is repeatedly performed reducing the number of words of the candidate text fragment so as to reach the best fit. The value used for the unique run delivered to organizers is 80% for the former direction. This threshold is increased by an additional 10% for the reverse direction. This value has been empirically set to provide best precision although sometimes sacrifying recall (see section 6.1).

Moreover, during development phase, other methods for SF-EF alignment have been used, like guessing based on textual and/or linguistic cues or allowance of wider intertwined gaps not constituting part of the final SF. However, the most important of them is described below.

If so demanded, the alignment method can take advantage of Google Translation Services. There is no free (or trial) license for Google's API Cloud Translation. Therefore, we have developed a wrapper over the HTTP web service, being then the issue that queries must be time elapsed in order not to be banned by Google. Thus, processing is kept asleep during enough time for the set of queries not to be blocked but, since these are synchronous to the whole process, the strategy has only been used during development, in order to possibly populate both Spanish and English EF attributes. The method uses a disk cache that can be inspected before the information contained in it is converted to the XML lexicon format.

We illustrate this with one of the SF candidates occurring in the background dataset —"NSE". No other previous matching method (not even permutations) applies for the resolution of the pair "enolasa específica neuronal (NSE)". However, the system, in this situation, launches a query to the aforementioned service with the previous string[10]. The translation service returns the English translation "neuron-specific enolase", which happens to match the Initials RE[11]. Note, however, that the text fragment "enolasa neuroespecífica NSE", which, in any case, includes a SF that is not spotted by the program, would not have produced a correct alignment since Google's translation for this variant is "neurospecific enolase" and, hence, Initials RE wouldn't match.

As regards nested structures, also included in task 1 specifications, ARBOREx only implements the case where all but the last character of two consecutive SFs match, as it is shown in the following example: "desarrollo de retracción palpebral superior de 3 mm en ambos ojos e inferior de 2 mm en ojo derecho (OD) y 1 mm en el izquierdo (OI)". For these structures, ARBOREx forces that both

[10] We won't go into the details on how we set the correct text spanning for these cases.

[11] This is the unique RE used by the program in the case of dynamically translated EFs. The hyphen is considered a word boundary in this type of REs.

candidate EFs match (through ASM) with the corresponding lexicon EF.

As a final comment, the alignment method takes as argument a hash of flags allowing the tailoring of the whole process depending on the scenario.

ARBOREx processes this task at 5,000 word/second[12] on an Intel Core i7-3770K @3.5GHz, running Ubuntu 16.04.3 (Xenial).

5 Task 2

For this task, BARR2 specifications asked participants to extract all SFs mentioned in each clinical text and try to predict their correct EF depending on the context. The challenges in this task don't differ from those of general language processing since homonymy/polysemy and synonymy are the main barriers that have to be overcome. To be more precise, besides the treatment of spelling variants for SFs, systems have to deal with homonymy issues given that most of the SFs may have multiple EFs. In this sense, the task is similar to the lexical sample task for WSD tracks like SENSEVAL-2 [5]. There are also issues concerning synonymy and language equivalent EF prediction, as well as the abandonment of linguistic properties of EFs (linguistic form, lemmatization, stop word deletion) in favor of conceptual ones that will be described in future work[13].

Clinical texts characterize as a lexically very complex discourse type. Hence, the tokenization with traditional NLP tools (pipelines, libraries and the like) is problematic, since some effort has to be devoted to adapt tokenization rules to this text type. However, syntactically, as it happens in any other genre, words are not thrown randomly like dices onto the text. On the contrary, clinical texts show a very fixed set of predicates, patterns, expressions that are the backbones on which clinical record components like patient's medical history, current symptoms, diagnostic, medication history, treatment, test results (hemogram, imaging,...), prognosis can be induced. Obviously, it is through a proper clinical record segmentation that the best linguistic and terminological processing of these texts can be done —as implemented by, for instance, Demner-Fushman *et al.* [6]—, but, even without this segmentation, some hints can be used to point in the direction of the correct EF given a hypothesized record component. Seen like this, we are not far from the Firthian statement of knowing a word by the company it keeps. This very simple principle has been the driving force of the strategy described below.

In order to have a glimpse on the type of discourse the system had to work on, we carefully read 500 clinical case texts —400 from our various corpora and 100 more from BARR2 datasets. The cases where randomly selected with no particular distribution. During the reading process, we collected words, patterns, expressions and other linguistically relevant features. The information gathered was used as an aid for the design of the formalism described below as well as a

[12] As counted by linux command `wc`.

[13] Although see organizers' proposed solution to most of these issues in note 14.

means upon which conceptual classification work was started.

From the processing point of view, task 2 is built on top of task 1 since, prior to any SF spotting, ARBOREx performs all the SF-EF alignments found in every clinical case text. Besides, a previously computed SF-EF alignment table for each particular file is consulted. These tables are the result of processing the full paper where the clinical case is included in for potential (local) SF-EF alignments. Both alignment types are then used to mark —but just mark, with no disambiguation yet— a particular EF for each SF found in the clinical case.

The clinical case processing restarts then using every (either lexically or contextually) known SF, that is then spotted and cached (along with its offset and a contextual window) for EF resolution. For the actual EF prediction two strategies have been implemented: a rule-based approach based on `RE`s expressed in a clear formalism that simplifies rule writing, and a very simple bag-of-words model. Both strategies are described below.

5.1 Rule-based resolution

Albeit no previous record segmentation in its fundamental components was performed, as explained above, we have tried to map linguistic form to meaning using all heuristic knowledge at our disposal, even without any linguistic processing as such. This knowledge had to be expressed at the character level, since no tokenization was done on the text to be processed. In order to add some 'sugar' to the expression of such statements on form and possible correlation with meaning and, thus, to the prediction of the correct EF for each given known SF, we developed a basic rule formalism that allows the definition of certain abstractions and notational simplifications on the input stream. Besides, the formalism allows for a simple connection with the information stored in the system lexicon. In record time, we have built a (very incomplete) grammar for the resolution of EFs given SFs and a local context. Grammar syntax is not fully (neither formally) defined here, but the general format of rules is exemplified below along with some useful information on the way it works. The general rule format is as follows:

```
[lc] - [f] - [rc] > [x=zzz]
```

where `lc` and `rc` stand for left and right contexts (that can be optional), respectively, and `f` is the focus, usually corresponding to a SF (or a disjoint of several such SFs, but not `RE`s). Context slots are (simplified) `RE`s. As for the right hand side of the rule, `x` can be any of `m`, `c` or `a`, to match a given EF (possibly a `RE`), unify with a given lexical `concept` value, or add a EF, respectively.

Rule qualifiers also exist to, for instance, allow a unique context specification to be matched either to the left or right of the focus SF(s), handle typographic case by the user (it is handled 'intelligently' by the program by default), and/or establish the maximum distance of the context `RE` with respect to the focus. Besides, macros may also be defined in the rule file. The following fragment illustrates some of the features of the formalism:

```
def C_FINDING::PRED = diagnóstic|confirma|reali[zc]|descarta|
                      se sospech|desarroll|complic

p:[{{C_FINDING::PRED}}] - [{{l:C_FINDING}}] - > [c=C_FINDING]
```

In the example above, after defining a (simplified for this paper) symbol that tries to capture typical predicates introducing diseases, expressed as pseudo-lexemes if so desired given that there is no tokenization, a 'template' rule is declared. Double braces indicate a symbol to look up in the symbol table. In the focus, when the symbol starts with `l:`, the value is resolved as the disjoint set of the abbr values for entries whose concept type information is, in this case, C_FINDING (clinical finding). The RHS of the rule selects precisely the EF for entries having this concept value, using, if available, the preferred variant, as shown in section 2.2[14].

As an illustration, in the text fragment "Con el diagnóstico de ACG se inició tratamiento con Prednisona 1 mg/kg/día.", ARBOREx must spot "ACG" as a SF. It is included in our lexicon with two possible EFs —"angiocardiograma" and "arteritis de células gigantes". Being the latter coded as a disease and using the above rule, it is selected as the predicted EF.

Rules are applied in the order they appear in the grammar file. Once a rule is applied the prediction is considered finished, and the cursor moves to the next SF to resolve.

The grammar so built had 30 template rules and about 130 other rules when used on the BARR2 test dataset. For instance, there are rules that try to model SFs usually occurring in an hemogram or are part of a medication treatment. With a bit more effort and some testing —only one day of testing was devoted for the track—, we consider this a promising avenue for EF prediction in clinical texts.

5.2 Bag-of-words EF resolution

Although, as was stated in section 3, there is no tokenization —in the NLP sense— as part of the process, for this resolution method the system performs a naïve tokenization by splitting text on spaces and trimming all punctuation. ARBOREx then builds a set of n-gram lists (for $1 \leq n \leq 3$). Only unigrams have been used for this task. The content-word list for each EF for a given SF is then compared to the frequency profile for the clinical case text. The best scoring EF is then selected. If there is a tie at the highest score (that could be 0 if no word

[14] This was forced by BARR2 organizers since, although all the different EFs in a lexical entry are inter-language (reciprocal translations) or intra-language (variants and/or synonyms) equivalents, only one of them was selected during manual annotation. To somehow approximate Gold Standard attributions to system predictions, BARR2 organizers are in the process of manually mapping EFs to concept identifiers from "widely used medical and biomedical terminological resources and databases (incl. SNOMED-CT, UMLS and UNIPROT)." [e-mail communication from organizers to participants].

from any of the EFs occurs in the text), and there is no local SF-EF alignment provided in previous steps of processing, no EF is predicted for the given SF, hence favoring precision over recall.

Once more, the preferred value is used and can do the job with false ambiguities corresponding to variants of the same entry —that correlate to a single concept.

ARBOREx processes this task at 1,800 word/second with the same configuration as above.

6 Results and discussion

In this section we present the results obtained by our system in both tasks.

Given the approach adopted, we spent all the available time for resource (and tool) development based on the available manually annotated material in the training and development datasets. Thus, only one run was sent to organizers for the system to be evaluated. Organizers' evaluation method is based on the common precision, recall and F1 score metrics calculated on a subset of 220 clinical cases taken from the background dataset as manually annotated by domain experts. Table 1 shows the results obtained by our system for task 1.

	Precision	Recall	F1-score
Task 1	88.61	88.23	88.42

Table 1. Results for ARBOREx on BARR2 test dataset for task 1.

As it can be observed, ARBOREx is reasonably balanced between precision and recall for this task. The methodology adopted, that is backed-up by the abbreviation lexicon, allows for a good performance in the task. At the time of writing, BARR2 organizers have not released results for all the participants in the track. However, compared with the BARR first edition best performing system for abbreviation relation task, which obtained an F1-score of 67.74%, our results improve this record by 20 points.

6.1 Error analysis for task 1

We have performed error analysis in order to have a clear idea of the best way to improve the system. For task 1, the organizers' test subset produces 238 relations. The comparison of these with our run is shown in table 2. Note that apart from exact alignment between the sources, we also include cases where only a minor defect has been found in either of the versions. For these cases, the prefix M indicates manual version (Gold Standard) whereas A stands for automatic version.

		Golden	ARBOREx
Total		238	237
Aligned		210	210
A-Excess		–	6
A-Defect		6	–
M-Defect		–	4
Semi-aligned	**A-Longer SF**	2	
	A-Longer EF	1	
	A-Shorter EF	1	
	M-Longer EF	2	
	M-Shorter SF	1	
Nested		14	8
A-Disputed Case		–	1
Common Disputed Case		1	

Table 2. Error analysis for task 1.

At first sight, it can be observed the high impact of nested structures on the overall differences. This is, in fact, a major drawback of nested structures representation in the output file. We haven't gone into the details of the evaluation program developed by the organizers but, according to the program description, the program tries to align —line by line— the prediction file with the Gold Standard file. Since each SF-EF pair in a nested structure is represented using two different lines, wrong cases weigh more than errors in non-nested pairs. The differing figures in each column is a consequence of many-to-one (or many-to-none) relations between both files. Anyway, the high figures reveal that there are many more cases of this type of structure than in non-technical discourse. Hence, this is an area for further development of the program.

A second relevant finding is that silence (unannotated true pairs) affect not only to the prediction system but also to the manually annotated dataset[15]. These M-Defect cases —that are recorded in order for the Gold Standard to be corrected— include the pair `<HSA, Hemorragia Subaracnoidea>`, which occurs twice in one of the files, while in the Gold Standard only occurs once. This is also the case for the pair `<AV, agudeza visual>`, where only the first occurrence in a file has been annotated in the Gold Standard. Besides, the pair `<SCID II, Entrevista Clínica Estructurada para los Trastornos del Eje II del DSM-IV>` is a valid pair not manually annotated[16]. Finally, the pair `<EMA, antígeno de membrana epitelial>` is missing in the Gold Standard, albeit occurring in one of the test files.

[15] Figures for this type and also for A-Excess and A-Defect are swapped on purpose in the table so the sum calculation is clearer.

[16] The SF corresponds to the English EF "Structured Clinical Interview for DSM-IV Axis I Disorders".

Two of the errors in A-Defect type have a unique capital letter as SF —a wordform that is very difficult to capture: “R” and “H”. Other has to do with ASM threshold used for the run (80% for the zoned text containing the lexical EF). With this value, the lexically coded EF “inmunoglobulina intravenosa” has a similarity below this value with respect to “inmunoglobulinas endovenosas”, the actual string in the text. Another two cases have to do with presentation contexts not covered yet by the system or variants not obtained through Google’s Translation Services —“enolasa neuroespecífica”, as commented on footnote 11. The last A-Defect error shows a shorter SF error in the Gold Standard, as the tuple `<intervalo, intervalo entre ondas P y R>` was attributed (SF should be “intervalo PR” in this case).

We have also performed a glassbox analysis of semi-aligned cases —those where there is a minor difference between both sources, either in the SF or the EF values. There are 7 such cases for the various possible situations. Once more, we detail manually annotated ones for dataset improvement. As regards M-Longer EF type, the Gold Standard includes the tuple `<RMN, resonancia magnética cerebral>`, where the last word is not part of the expansion of the abbreviated form, and the tuple `<BAS, broncoaspirados de secreciones>`, where the modifying PP does not belong to the EF of this abbreviation. The M-Shorter SF error is for the tuple `<ARA, antagonista del receptor de angiotensina II>`, in which “ARA II” should be the preferred SF.

Our system extended the spotted SF in two cases: “MEN 1” —the digit not belonging to the SF—, and “WAIS-III” —again, with the Roman number not being part of the SF. It also extended the EF in the case of “prueba fluorescente de anticuerpos contra el trepanoma” —with the noun “prueba” not being part of the EF. Finally, it shortened “epitelio pigmentario de la retina” for SF “EPR”, not extracting the PP. This case is interesting, since the abbreviation is included in the *DSM* with the EF “epitelio pigmentario retiniano”. The strict similarity threshold used in the run validates the equivalence in the lexicon-text but not in the opposite direction. Anyway, Initials `RE` should have been applied prior to allowing multiple characters in a single word. This behavior seems a bug in the program, something that also happens with a couple of other non-sense predictions.

Another difference between attributions and predictions is the Common Disputed case in which the Gold Standard validates the pair `<HHV, virus herpes humano tipo 8>` where the system prediction is `<HHV, virus herpes humano>`. Note that in order to be correct with the attributed EF, Golden Standard should read `<HHV 8,...>`. Finally, the A-Disputed Case is exemplified by the following context: “mediante tomografía de coherencia óptica (Stratus OCT;. . . ”, where the system outputs the tuple `<OCT, tomografía de coherencia óptica>`. This is arguably a wrong prediction, since “OCT” is part of a product name in this context, but the SF-EF alignment is correct.

Task 2 was a new challenge for this year's BARR competition. For its successful completion, besides a good SF-EF lexical base, a mechanism for EF resolution, either induced from available training data or manually built from careful study of clinical cases, was needed. As probably all the other participants new to this domain, we had a very strict deadline to implement (and test!) such a piece of software. In our case, we spent most of the available time (once task 1 was developed to a subjectively satisfactory level) in defining and implementing the formalism to simplify our RE grammar rules. Hence, the time spent in grammar creation was very limited and testing was performed on a subset with only 100 clinical cases. Nevertheless, results for this task, as provided by the organizers, are very promising. Unfortunately, track organizers did not release all system results, so nothing can be commented on our results as compared to other systems, being this, as has already been explained, a new task. The results for this task are shown in table 3.

	Precision	Recall	F1-score
Task 2	88.34	81.11	83.17

Table 3. Results for ARBOREx on BARR2 test dataset for task 2.

Once more, results show a balance between precision and recall. Recall is lower this time because, for those cases where there are multiple EFs for a spotted SF, if no rule was applied and there was a tie in weighting among the multiple EFs, we simply didn't output anything, although selecting the most frequent EF in the training/development/background sets would probably had increased recall without affecting precision significantly. However, lack of time for experimentation on this strategy led us to take the conservative decision just described.

No error analysis has been performed for task 2. The reasons for this lack of analysis are twofold: on the one hand, the Gold Standard set for this task has 3,414 pairs, which increases the size of that of the task 1 by an order of magnitude; and, on the other, organizers have committed to map EFs to one of the widespread medical terminological resources, which probably will simplify the error analysis. Moreover, the latter improvement over both the Gold Standard dataset and the predictions file will produce a result table different (probably fairer) from that shown above.

To give just a couple of examples of the foreseen improvement both on the Gold Standard and the system evaluation, take for instance the DSM entry for "TAC". In this dictionary, the EF for this abbreviation is "tomografía axial comput**a**rizada". This SF occurs 92 times in the test dataset, 76 of them with the DSM EF but the other 16 with the variant "tomografía axial comput**e**rizada". The lemmatization column provided by organizers to improve comparability conditions simply doesn't work for cases like this. However, the proposed map to standard concept IDs makes this asystematic annotations irrelevant. As a second and final example, the training dataset included both “polymerase chain reaction” and “reacción en cadena de la polimerasa” (its Spanish equivalent) as EF for the SF “PCR”. We decided to use the Spanish EF, just to discover that in the Gold Standard the English one was the only used in the 6 cases it occurs in this subset[17]. Thus, there is room for improvements for both the prediction system(s) and the manually annotated dataset(s).

7 Conclusions

We have presented an abbreviation identification tool for Spanish medical texts specially tailored for clinical reports, ARBOREx. Although firstly conceived for the alignment of SF-EF explicit relations in a text (task 1 in BARR2 competition), its performance seems also very promising for the resolution of (possibly ambiguous) EFs when no explicit reference to them exist in the text (task 2). We are currently working on debugging and code optimization, better expressive mechanisms for the formalism developed, and specially on the extension and enrichment of the resources used by the system.

References

1. Intxaurrondo, A., Marimon, M., Gonzalez-Agirre, A., Lopez-Martin, JA, Rodriguez Betanco, H., Santamaría, J., Villegas, M. and Krallinger, M. Finding mentions of abbreviations and their definitions in Spanish Clinical Cases: the BARR2 shared task evaluation results. SEPLN (2018).
2. Islamaj Doğan R, Comeau DC, Yeganova L, Wilbur WJ. Finding abbreviations in biomedical literature: three BioC-compatible modules and four BioC-formatted corpora. *Database: The Journal of Biological Databases and Curation.* 2014;2014:bau044. doi:10.1093/database/bau044.
3. Intxaurrondo, A., de la Torre, JC, Rodriguez Betanco, H., Marimon, M., Lopez-Martin, JA, Gonzalez-Agirre, A., Santamaría, J., Villegas, M. and Krallinger, M. Resources, guidelines and annotations for the recognition, definition resolution and concept normalization of Spanish clinical abbreviations: the BARR2 corpus. SEPLN (2018).
4. Yetano Laguna, J., Alberola Cuñat, V. *Diccionario de Siglas Médicas.* Ministerio de Sanidad y Consumo, Madrid, Spain (2003).
5. Preiss, J., Yarowky, D. (eds.). *Proceedings of SENSEVAL-2, Association for Computational Linguistics Workshop*, Toulouse, France (2001).
6. Demner-Fushman, D.; Abhyankar, S.; Jimeno-Yepes, A.; Loane, R.F.; Rance, B.; Lang, F-M; Ide, N.C.; Apostolova, E.; Aronson, A.R. A Knowledge-Based Approach to Medical Records Retrieval, *Proceedings of TREC* (2011).

[17] Actually, there is a typo in 5 of the 6 cases that needs correction before concept mapping. This typo is the one highlighted in figure 1.