A proposed system to identify and extract abbreviation definitions in Spanish biomedical texts for the Biomedical Abbreviation Recognition and Resolution (BARR) 2017

Christian Sánchez, Paloma Martínez

Computer Science Department, Universidad Carlos III of Madrid Avd. Universidad, 30,
Leganés, 28911, Madrid, Spain

Abstract. Biomedical Abbreviation Recognition and Resolution (BARR) is an evaluation track of the 2nd Human Language Technologies for Iberian languages (IberEval) workshop, which is a workshop series organized by the *Sociedad Española del Procesamiento del Lenguaje Natural (SEPLN)*. In this first edition of BARR, the focus is on the discovery of biomedical entities and abbreviation, and relating detected abbreviations with their long forms. This paper describes the system and the approach presented in this track. We develop a ruled-based system using an adapted version of the algorithm for extraction of abbreviations and their definitions from biomedical text proposed by Schwartz & Hearst.

1 Introduction

The interest in automatic text processing of biomedical documents has increased in the last few years. There is some research that proposes certain solutions and approaches for the problem of recognition and resolution of abbreviations, acronyms and symbols, but most of it is focused on analyzing text written in English. In this context, the Biomedical Abbreviation Recognition and Resolution (BARR) track asked the participants to test and submit processing text systems that are able to analyze and extract occurrences of abbreviation-definition pair in biomedical documents written in Spanish.

For our approach, and in line with other previous work and research, we refer to an abbreviation as a *Short Form (SF)* and the definition as the *Long Form (LF)*. For this track, beside the mentioned elements, if a Short Form appears again in the text it is considered or marked as Multiple, and if it is only an abbreviation without a definition it is marked as Global.

This paper is organized as follows: Section 2 describes our proposed approach. Section 3 presents evaluation and results. Finally, conclusions and future work are discussed in Section 4.

2 Proposed Approach

As previously stated, our goal was to analyze documents written in Spanish in order to extract abbreviations and definitions. To accomplish this, first, we assume that a Short Form will be placed after the Long Form, i.e: [5]

Las secuelas en la articulación temporomandibular (ATM)

In this example we consider ***ATM*** to be the abbreviation (*Short Form*) and ***articulación temporomandibular*** its definition (Long Form). The documents used for the track contain a title and abstract; the analysis should be done in both.

Our system is based on the algorithm proposed by Schwartz et al., which is divided into two main tasks. The first one is to identify the Short forms [5]:

> Short forms are considered valid candidates only if they consist of at most two words, their length is between two to ten characters, at least one of these characters is a letter, and the first character is alphanumeric.

The second task is to identify candidates for the long forms. As explained in the mentioned paper, the long form must appear in the same sentence as the short form, it should have no more than min(—A— + 5, —A— * 2) words, and composed of contiguous words from the original text that include the word just before the short form. When these tasks are completed, the main idea of the proposed algorithm is [5]:

> Starting from the end of both the short form and the long form, move right to left, trying find the shortest long form that matches the short form. Every character in the short form must match a character in the long form, and the matched characters in the long form must be in the same order as the characters in the short form. Any character in the long form can match a character in the short form, with one exception: the match of the character at the beginning of the short form must match a character in the initial position of the first (leftmost) word in the long form (this initial position can be the first letter of a word that is connected to other words by hyphens and other non- alphanumeric characters).

Using the mentioned statements as guidelines, we divided the system process into the following tasks:

2.1 Short Form Identification and Validation

The Short Form identification is performed using a set of rules and regular expressions. The title and abstract are analyzed individually to get all the matches for any term which complies with the criteria. Also its position in the text its extracted. At this point the validation is performed with the use of regular expression and some pattern rules. One of the regular expressions used was the following:

```
([A–Z]{2,}[\−\/A–Z0−9]∗)\b
```

It matches terms which contain at least 2 uppercase letters and could contain the symbols (-, /) or any number. For example, in the following text:

Antecedentes y objetivos. El Registro Informatizado de Enfermedad Tromboembólica (RIETE) es un registro prospectivo que incluye de forma consecutiva pacientes diagnosticados de enfermedad tromboembólica venosa. Hemos comparado la presentación clínica y la respuesta al tratamiento anticoagulante en pacientes con enfermedad tromboembólica venosa idiopática (ETEVI) versus secundaria (ETEVS, asociada a algún factor de riesgo). Pacientes y métodos. Se analizaron las diferencias en las características clínicas,

The system will match the terms: *RIETE, ETEVI*, and *ETEVS*.

Also, and in difference with the based Schwartz et al. work, we estimated the length of an abbreviation between 2 and 8 characters and used that as a pattern rule. To determine this length we use as a reference a resource which contains Spanish Medical Abbreviations: *Diccionario de siglas médicas* [1]. This estimation was made after an analysis of the **3386** terms contained in the dictionary. First we obtained those terms formed just by a word. This query matched **2676** terms, but in the results we got words like: *'Arterioesclerosis'* which is not an abbreviation but a name of a disease. Then we change the query to match all the terms starting with a least 2 capital letters. Applying this criteria the documents matched were **2319** and we obtained terms like: 'AA' which have a few definitions *(Aminoácido, Anemia aplásica, Aorta abdominal)* or one of the longest *'PETHEMA'* (*Programa para el estudio de la terapéutica de las hemopatiías malignas*).

2.2 Short Form Extraction

Once the terms are identified, they are stored in two lists (title, abstract) with their positions in the text. Using the same text for the example above, the following demonstrates how the system stores the results for the abstract analysis:

```
[
[0]: {
        positions:   [
                [0] "82:87"
                ],
        term:          "RIETE"
},
[1]: {
        positions:   [
                [0] "358:363",
                [1] "626:631",
                [2] "729:734",
                [3] "798:803",
                [4] "1130:1135",
                [5] "1302:1307",
                [6] "1729:1734"
```

[1] http://sedom.es/diccionario/

```
                ],
                term:           ”ETEVI”
        },
        [2]: {
                positions:    [
                        [0] ”384:389”,
                        [1] ”634:639”,
                        [2] ”755:760”,
                        [3] ”865:870”,
                        [4] ”1001:1006”,
                        [5] ”1111:1116”,
                        [6] ”1253:1258”,
                        [7] ”1765:1770”
                ],
                term:           ”ETEVS”
                },
        ]
```

As shown above, multiple matches of the same term are grouped as one record in the list. This allows one to identify how many different terms are in the text. An abbreviation could appear many times in a text. In this case the abbreviation that appears together with a definition or *Long Form* is marked as the *Short Form* and the others are marked as *Multiple*.

2.3 Long Form Identification and Validation

The Long Form identification is performed after the extraction of the Short Forms. Here a similar approach as proposed by Schwartz is used. We take the position of each term and evaluate the text from right to left. Each character of the term is used to find, for instance, a word which starts with the same letter. If it does not match, a search inside the word is performed to check if the word contains the letter. Stop Words, numbers or any other non alphanumeric character are not take into account for this evaluation. Once the number of matches in the text is equal to the length of the Long Form, the system considers that it has a set of candidate words for the Long Form. The following is an example for the extraction of the candidate words for the term ETEVI:

```
        {
                0:    {
                        I :       1,
                        word:     ”idiopatica”
                },
                1:    {
                        V:        1,
                        word:     ”venosa”
                },
            2:    {
                      E:        1,
                      T:        1,
                      word:   "tromboembolica":
            },
            3    {
                      E:        1,
                      word:    "enfermedad"
            }
    }
```

Here, each letter of the term has a matched word. Notice that in the element with the index '2', the word "tromboembolica" has two matched letters: the letter (t) at the beginning and (e) in the middle.

2.4 Long Form Extraction

With the list of word candidates, the system identifies the start position of the leftmost word candidate and the end position of the rightmost candidate and extracts all of the text contained between them. With this we can obtain also the stop words discarded in the identification step. Once a Long Form is extracted the term is marked as the Long Form and the other occurrences are marked as Multiple. If there were not candidate words the system classified the term as Global.

3 Evaluation and results

For the BARR track, the participating systems are evaluated with the F1-micro measure. There were two required submissions to evaluate : entity prediction and relation prediction. Before evaluation the system was tested with the sample corpus to get a glimpse on how to process the data. At first we noticed that the files needed to be treated with an UTF-8 encoding, as is recommended when processing documents written in Spanish or other languages different than English. Here lies the main difference with our approach and the one proposed by Schartz et al., because this subject is key for proper data extraction and manipulation for this track.

The main programming language for the system is Perl. For the first processing test a comma-separated values file manipulator module was used: Text::CSV [2], mainly because it could be easily configured to open and process tab-separated files, which is the format for the datasets provided by the track. This module converts the bytes to UTF-8 character equivalents by default. This behavior is fine for display compatibility on different operative systems and language configurations, but it was identifying the abbreviations and definitions in different positions than the labeled data provided by the organizers. This is an example from the labeled sample dataset:

1741 es A 19 49 articulación temporomandibular LONG

[2] https://metacpan.org/pod/Text::CSV

When the system processed the same text, the followed result was obtained:

1741 es A 19 50 articulación temporomandibular LONG

Here, the last position is different due to the decoding of the accented vowel ***á***, an extra byte is added to create a valid UTF-8 char sequence. This behavior is explained in the module documentation. One can deactivate this option and the bytes (and therefore the length of the string) will not change. The results obtained for the system matched with the labeled sample data, but a different issue emerged, as is shown in the following example:

1741 es A 19 49 articulaci¡F3¿n temporomandibular LONG

Here the string stored in the file is not displayed correctly. The module tries to convert the bytes to a valid ASCII character to be printed. In this case it is not possible to convert the UTF-8 byte, then is replaced with the symbol ¡F3¿. The solution was to relay in the Unicode encoding/decoding methods provided by the language itself. Perl has the capacity to handle Unicode natively ([3]). We discarded the use of the module mentioned before and handle all the file parsing and processing with the native I/O file methods, divided every record in the dataset into fields (id, language, title, abstract) to obtain the text to be processed. After this adjustment, the results were displayed properly:

1741 es A 19 49 articulación temporomandibular LONG

Another encoding related issue we had to fix, were cases when a Long Form contained vowels with accent in the candidate words. An example that illustrates this problem is the case with the term *DMO*. The system extracted this term in the following text:

densidad mineral ósea (DMO)

The Long Form detection did not match the word *ósea*. The fix was to convert the vowel in their not accented equivalent before the evaluation.

Once the mentioned issues were fixed, we could generate the entities and relations predictions required by the track. To test the system performance we run an evaluation with the sample dataset to compare the results with the other baselines used at the track. We show the Entity and Relation Evaluation results compared in Table 1 and Table 2 respectively .

Tool	Precision	Recall	F-Measure
Ab3P	78.20	39.87	52.81
ADRS	70.75	49.02	57.91
BADREX	72.50	37.91	49.78
Our System	83.67	53.59	65.33

Table 1 Results from Entity Evaluation

[3] http://perldoc.perl.org/perluniintro.html

Tool	Precision	Recall	F-Measure
Ab3P	71.79	34.14	46.28
ADRS	62.26	40.24	48.89
BADREX	52.38	26.83	26.83
Our System	55.38	43.90	48.98

Table 2 Results from Relation Evaluation

Our system did not get good results for the Relation Evaluation, but it got best results compared with the others in the Entity Evaluation. The results for the final submission are presented in Table 3.

Evaluation	Precision	Recall	F-Score
Entity	70.69	73.47	72.05
Relation	72.20	61.78	66.59

Table 3 Final Submission Results

4 Conclusions and future work

In this paper we presented a rule based system for automatic detection and extraction of abbreviations and their definitions. With the use of some pattern rules and regular expressions and adapting a former proposed algorithm, we attained some results that could be improved. Using Perl for text parsing provides a good performance. Using the test dataset (20000 records) on a MacBook Pro (Retina, 13-inch, Late 2013) with a 2,4 GHz Intel Core i5 CPU and 8GB RAM, the estimated time to generate the entity prediction file was 55s and for the relation prediction was 46s. One thing to notice is that our system did not detect terms derived from Short Forms nor the Nested Relations, which were one of the requirements for the predictions. This should be one of the possible additions to improve the system.

When working with Spanish text, it is important to take into account the encoding of the files. It could be a critical issue if it is not handled correctly. We noticed that after implementing this helped to improved our results.

The system needs to be improved to detect Long Forms that could be at the right of the abbreviation for example, given the following text:

ATM (Las secuelas en la articulación temporomandibular)

In this case, and contrary to our assumption in handling documents, the detection pattern is *Short Form (Long Form)*. There is room to improvement or different applications. Examples include using Machine Learning to classify abbreviations into different tags in order to obtain the category of a document or as well another use case, generating abbreviations (short form) from a candidate definition (long form) and validate it using a dictionary.

References

1. Intxaurrondo, A., Pérez-Pérez, M., Pérez-Rodríguez, G., Lopez-Martin, J., Santamaría, J., de la Peña, S., Villegas, M., Akhondi, S., Valencia, A., Lourenço, A., Krallinger, M.: The biomedical abbreviation recognition and resolution (barr) track: benchmarking, evaluation and importance of abbreviation recognition systems applied to spanish biomedical abstracts. (SEPLN 2017)
2. Krallinger, M., Intxaurrondo, A., Lopez-Martin, J., de la Peña, S., Pérez-Pérez, M., Pérez-Rodríguez, G., Santamaría, J., Villegas, M., Akhondi, S., Lourenço, A., Valencia, A.: Resources for the extraction of abbreviations and terms in spanish from medical abstracts: the barr corpus, lexical resources and document collection. (SEPLN 2017)
3. Manabu Torii, Z.z.H., Song, M., Wu, C.H., Liu, H.: A comparison study on algorithms of detecting long forms for short forms in biomedical text. BMC Bioinformatics (2007)
4. Okazaki, N., Ananiadou, S.: A term recognition approach to acronym recognition. Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions pp. 643–650, (2006)
5. Schwartz, A., Hearst, M.: A simple algorithm for identifying abbreviation definitions in biomedical text. Pacific Symposium on Biocomputing pp. 451–462 (2003)