BARR task baselines

Martin Krallinger[1], Santiago de la Peña[1], Ander Intxaurrondo[1], Jesus Santamaria[1], Jose A. Lopez-Martin[2], Alfonso Valencia[3], Marta Villegas[3], Analia Lourenço[4]
[1]Biological Text Mining Unit, Spanish National Cancer Research Center, Madrid, Spain
{mkrallinger,sdelapena,aintxaurron,jsantamaria}@cnio.es
[2]Medical Oncology, Hospital 12 de Octubre, Madrid, Spain
jalopezmartin@gmail.com
[3]Life Sciences and Computational Biology, Barcelona Supercomputing Center, Barcelona, Spain
{alfonso.valencia,marta.villegas}@bsc.es
[4]Next Generation Computer Systems Group, University of Vigo, Spain
analia@uvigo.es

1 Introduction

The Biomedical Abbreviation Recognition and Resolution (BARR)[1] track promotes the development and evaluation of biomedical abbreviation identification systems within abstracts of biomedical documents written in Spanish. The BARR track requires both the recognition of short form (*SF*) and long form (*LF*) candidate pairs from sentences, and identification of exact string boundaries.

This document describes the abbreviation detection tools used to create the baselines of each track at the BARR evaluation task, and how we generated each baseline.

2 Evaluation tracks

In this section, we introduce the two tracks of the task.

2.1 Entity evaluation

In this task, the participants must detect biomedical entities in the corpus, both long forms and abbreviations included.

2.2 Relation evaluation

In this task, participants must associate the abbreviations (or short forms) found in the corpus with their long forms. Both must be in the same context.

3 Tools

This section describes the different open-source tools we used to detect abbreviations in the corpus. These 3 tools work using simple regular expression rules to detect abbreviations and their long forms in the text. None of them specifies the offsets of the long and short forms.

These tools have been previously tested in English corpora, but they work well with Spanish biomedical publications. None of them uses internal abbreviation dictionaries.

To get the results, it is recommended to use a sentence splitter beforehand, and apply the following tools sentence by sentence. We used IXA Pipes (Agerri et al., 2014) to split sentences.

3.1 Ab3P

Ab3P (Abbreviation Plus Pseudo-Precision) is a simple tool developed by (Sohn et al., 2008)[2]. It is developed in C++, and the compilation process is quite simple.

The software outputs short forms and their long forms detected in the sentence, together with the estimated precision.

3.2 ADRS

ADRS (Abbreviation Definition Recognition Software) is another simple tool developed by (Schwartz and Hearst, 2003)[3].

To make use of this software, you just need to pass the file's path you want to analyze. The system will return short and long form pairs.

[1] http://temu.inab.org/

[2] https://github.com/ncbi-nlp/Ab3P

[3] http://biotext.berkeley.edu/code/abbrev/ExtractAbbrev.java

Tool	Micro-precision	Micro-recall	Micro-F1
Ab3P	78.20	39.87	52.81
ADRS	70.75	49.02	57.91
BADREX	72.50	37.91	49.78

Table 1: Entity evaluation baselines. Sample set.

Tool	Micro-precision	Micro-recall	Micro-F1
Ab3P	71.79	34.14	46.28
ADRS	62.26	40.24	48.89
BADREX	52.38	26.83	35.48

Table 2: Relation evaluation baselines. Sample set.

3.3 BADREX

BADREX (Biomedical Abbreviation Expander) is a GATE plugin developed by (Gooch, 2012)[4].

Although you need may GATE to run the plugin, it is possible to make use of it with the API. The software extracts both short forms and their corresponding long forms.

4 Baselines

In this section, we explain how we created the baselines for each track. To get these baselines, we just executed the tools described in section 3 and adapted the outputs to the track's evaluation format.

The evaluation of these baselines was done with the sample set. We will follow a the same process for the final testing set.

4.1 Entity evaluation

For the entity evaluation track, we extracted the long and short forms detected by each tool. These tools return long and short form pair, so we just took the entities found in these pairs. Later, we detected the positions of each entity in the titles and abstracts, and extracted the offsets. Finally we assigned the *LONG* or *SHORT* category, specified in the outputs. Entities that are not part of any *SF-LF* relationships are labeled as *MULTIPLE*.

Table 4.1 shows the final results of the entity annotation track for each abbreviation detection tool.

4.2 Relation evaluation

For the relation evaluation track, we extracted the long and short form pairs detected by each tool. Once we had them, we analyzed the titles and abstracts to get the offsets of each entity, and finally created the file to evaluate.

We consider a *SF-LF* pair those which are very close to each other. In other words, both short and long form should be participating in the same context. If we find the long and short form at the beginning of the document, we make pairs with them; meanwhile, if the short form appears once again later in the document, in another sentence, we do not make pairs between the second short form and the long form.

None of these tools detect *NESTED* relations.

Table 4.2 shows the final results of the relation evaluation track for each abbreviation detection tool.

5 Conclusions

We presented the baseline results of the BARR evaluation task. The baseline is built using three simple open-source abbreviation recognition tools, which detect long and short forms using regular expressions. Although these tools perform well in the detection of *SF-LF* pairs, they have some limitations when extracting complex entities and entity pairs.

References

Rodrigo Agerri, Josu Bermudez, and German Rigau. 2014. Ixa pipeline: Efficient and ready to use multilingual nlp tools. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*.

Phil Gooch. 2012. Badrex: In situ expansion and coreference of biomedical abbreviations using dynamic regular expressions. *CoRR*.

A.S. Schwartz and M.A. Hearst. 2003. A simple algorithm for identifying abbreviation definitions in biomedical text. In *In Proceedings of Pacic Symposium on Biocomputing*.

Sunghwan Sohn, Donald C. Comeau, Won Kim, and W. John Wilbur. 2008. Abbreviation definition identification based on automatic precision estimates. *BMC Bioinformatics*.

[4] https://github.com/philgooch/BADREX-Biomedical-Abbreviation-Expander