International Journal of Computational Linguistics and Applications, Vol. 6, No. 2, 2015, pp. 177–192
Received 22/06/2015, Accepted 24/07/2015, Final 23/09/2015.
ISSN 0976-0962, http://ijcla.bahripublications.com

A Text Mining Library for Biodiversity Literature in Spanish

JUAN M. BARRIOS
ALEJANDRO MOLINA
RAUL SIERRA-ALCOCER
ENRIQUE-DANIEL
ZENTENO-JIMENEZ
The National Commission for Knowledge and Use of Biodiversity (CONABIO)

ABSTRACT

Biodiversity represents a great ecological, economic and aesthetic heritage to the world. Most of the knowledge about this heritage could be found in thousands of documents that describe valuable information obtained over centuries.

Projects which try to gather and structure all this information, even for very specific topics, may take years. In addition to this, keeping a project updated is difficult because new knowledge is continuously being published. Therefore, there is a necessity to use automatic methods to extract relevant information efficiently. In this article we describe the first stage of a software project, that aims to build a complete library to apply Natural Language Processing techniques on documents about biodiversity in Spanish.

1. INTRODUCTION

This project is part of a large and permanent effort at the National Commission for Knowledge and Understanding of Biodiversity (CONABIO) to gather information about biodiversity in Mexico. The mission of the CONABIO is to promote, coordinate, support and carry out activities aimed at the knowledge of biological diversity in Mexico, and its preservation

for the benefit of society. As part of this mission, CONABIO is invested in creating and publishing knowledge databases about Mexican biodiversity. Up to know, most of the efforts in this direction have been carried out through traditional literature review. Every new project represents a significant challenge because it requires to select and organize thousands of potentially relevant documents to a new topic and then to extract the information on those documents and structure it on databases.

Natural Language Processing techniques offer an opportunity to automatize some of the tasks involved in these projects, saving many person-hours and increasing effciency by orders of magnitude.

In this article we present the first delivery of a Text Mining library focused on the extraction of information about biodiversity from documents in Spanish. This is the first stage of a software project that will go from extraction of plain UTF-8 text from PDF/OCRed files to the automatic extraction of fragments about uses of biodiversity in Mexico.

The progress made during the first phase includes the following features:

- OCR/parsing of PDF files,
- parsing correction,
- sentence segmentation,
- traditional species uses extraction,
- indexing of named entities,
- efficient Global Names Recognition and Discovery service call,
- extraction based in lexical patterns,
- and multiprocessing.

The tools in this first phase will enable us to develop more complex modules. For instance, we are currently working in a data set to induce models for Named Entity Recognition focused on species. The development of this library is an open initiative licensing under GPLv2[1], and can be downloaded from the repository <https://bitbucket.org/conabio_cmd/text-mining>.

[1] http://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html

2. STATE OF THE ART

Some NLP tasks require to be adapted for application in biology. As a consequence, new NLP challenges have emerged thanks to the interaction with biosciences data. For instance, the Biodiversity Heritage Library is in the process of digitizing 600000 pages of text a month, making them available as pdf image les and OCR text files.[2] However, biodiversity literature can be especially difficult to OCR and the current rate of digitization prohibits manual correction of these errors. Proposed solutions include components of crowd-sourcing manual corrections for automated corrections [1].

Aside from text correction, domain terms extraction is also a common topic concerning NLP applied to biology. Named Entity Recognition efforts have been oriented to detect species names (taxon) using mainly two approaches: lexicon based (dictionaries) and machine learning based. Lexicon based approaches focus on finding words that are contained in dictionaries previously given to the computer. An example is Linnaeus, designed specifically for identifying taxonomic names in biomedical literature using pattern matching [2]. Taxon Finder detects scientific names by comparing the name to several lists [3]. Taxon Grab uses a combination of nomenclatural rules and dictionaries of non-taxonomic English terms [4]. Supervised machine learning approaches rely on providing substantial training examples to a system that would reproduce a specific task. In the case of taxon name detection, the letter combinations within the names as well as the context are helpful to recognize scientific names. NetiNeti, a supervised learning algorithm, based on Bayes conditional probability, uses these features [5]. NetiNeti may learn, for instance, that a word with the first letter capitalized and ending with “es” is probably a taxon name, even though that word has never appeared in training examples.

Concerning Information Extraction, very interesting applications has recently been proposed. In [6], authors describe an algorithm to learn rules to extract leaf properties from plant

[2] www.biodiversitylibrary.org

descriptions. Another example is described in [7], where authors try to match patterns relating proteins (X activates Y, Y is activated by X, Y was activated by X, etc.).

Nevertheless, the majority of ready-to-use tools are only for English; this is the main reason to start a new project for Spanish and using Mexican literature.

3. PDF PARSING AND OCR

A common technical difficulty when working with PDF files is that those files may not have a text layer, in that case, it is necessary to apply optical character recognition (OCR). Sometimes, a PDF file has a text layer, but it does not have the permissions to extract it automatically.

For all this, we have developed a controller based on three different PDF parsers to obtain plain text[3]: Apache PDFBox, Apache Tika[4] and pdftotext[5]. Thus, we have managed to get plain text in most of cases. However, PDF parsers not always offer good results. Perhaps given the complexity of format in names in Latin, tabular information, bibliographical citations, varied typography and text columns; all this being quite common in specialized texts.

Using the library, it is possible to apply all of the PDF/OCR parsers at once. For instance, from the command line, we would do:

```
# txtm.py PDFparserController --infile f.pdf --
parser_type all
```

The file `f.pdf` will be parsed and for each parser a directory containing the extracted raw text will be created. This is useful to evaluate the quality of different outputs.

[3] PDF parsers require Java Runtime Environment (JRE).
[4] © The Apache Software Foundation
[5] © The Poppler Developers

4. Correction of Sentences Frontiers, Abbreviations and Hyphens

Since the text obtained from the parsing of PDFs is aligned with the print view, it is necessary to make corrections to reconstruct sentences. Example 1 shows a fragment of text extracted in which line breaks are found to fit the print format. This kind of text segmentation is useless to apply even the most basic techniques of NLP. For instance, the detection of a specific syntactic pattern in a badly splited sentence is impossible. Other sources of error in text segmentation are hyphened words and abbreviations, which may be confused with end of sentence punctuation.

> Example 1. En el transcurso de la elaboración de esta obra se fueron sumando participantes, de manera que a su conclusión cuenta con 79 colaboradores pertenecientes a 19 instituciones tanto académicas, como gubernamentales y no gubernamentales (cuadro 1). El Estudio está conformado por...

It has been necessary to include a specialized module to correct hyphened words and to detect the borders of sentences. Hyphen correction is based on regular expressions; while sentence segmentation is based on supervised learning. Using thousands of sentences manually annotated from a general corpus in Mexican Spanish; we have obtained a 90% precise segmentation on general texts. Both, training and evaluation of sentence segmentation are done using a wrapper of apache OpenNLP Sentence Detector.[6]

Although some segmentation problems are mitigated, errors still persist since there are countless abbreviations in the biological domain that are not included in our general corpus examples; therefore may not be learned by our segmentation model. In this regard, we have integrated a segmentation model using an specialized corpus from the domain and focused on examples of abbreviations and citations.

---

[6] http://opennlp.apache.org/documentation/manual/opennlp.html

Using the library, it is possible to apply the complete correction. From the command line, we would do:

```
# txtm.py PreprocessingController \
--infile f.txt \
--opennlp_bin <path_to_opennlp_bin> \
--opennlp_mod        /path/txtmining/txtmining/resources/
models/es-iula.bin \
--correction_type all
```

5. COMMON AND SCIENTIFIC NAMES OF MEXICAN SPECIES

Identifying species names in biodiversity literature is critical for a number of applications in data mining. At the current stage of the project, this task is tackled by using a lexicon-based approach that follows some ideas exposed in [2] adapted to Mexican species. We considered also the basic rules of scientific names writing mentioned in [8]. The lexicon lookup strategy scans a given input document, looking for terms that match (1) a word from the genera list, e.g. *Abeis*; (2) a word from the genera list followed by a word from the species list, e.g *Abeis mexicana*; (3) a word from the genera list followed by a specific abbreviation of species, e.g. *Abeis* (*sp*., *ssp*., *subsp*., *nov*.); (4) the abbreviation of a genus followed by a word from the species list or a specific abbreviation of species, e.g. *A. mexicana*; or (5) a common name from the list of common names, e.g. *Abeto de Vejar*.

We have included, in our development list, around 1000 genera, 2600 species and 13000 common names. However, new names or slightly bad written names will not be found, no matter how exhaustive are the list. For this reason, we have integrated the tool described in section 6 for name discovery and resolution.

Dealing with species common names is far more challenging. The lexicon lookup strategy suffers from important problems: (1) when names are homographs of other nouns (like *bandera*, *baraja*); (2) when names are homographs of common use words (like *ni*, *mis*, *ya*, *lo*); and (3) when short names match long ones too (like *barba* in *barba de chivo* or *palo* in *palo de agua*). The third complication could be solved using length-sorted lexicons. However, the two first problems need more accurate algorithms

to be detected and disambiguated. In future versions of the library, this feature will be added. The current version allows to integrate easily custom lists of names, basically by adding a file. The command to create an index of names contained in a file using a lexicon is:

```
# python txtm.py ReIndexController \
--infile f.txt --regexp_file <path_to_regexp_file>
```

6. GLOBAL NAMES RECOGNITION AND DISCOVERY SERVICE

The Global Names Recognition and Discovery (GNRD) service is a tool to recognize scientific names based on TaxonFinder [3] and NetiNeti [5] names discovery engines. Found names are optionally resolved against a number of resources.[7]

TaxonFinder detects only scientific names. Given a text, it will scan through the contents and it will use a lexicon-based approach to identify which words and strings are Latin scientific organism names. It also detects names at all ranks, including species, genus and subspecies but does not detect common names.

NetiNeti detects scientific names using machine learning. The system estimates the probability of a label (whether a name is scientific or not) by given a candidate string along with its contextual information.

The final response of GNRD service will combine the advantages of both engines. However, the language of incoming content is determined using unsupervised language detection. If the language found is other than English, only TaxonFinder is used. Therefore, the resolver best performance is expected to be for English. Nevertheless, it also shows enough accuracy for Spanish; considering that names can be optionally resolved by using some other certified resources. Our solution is only to consider certified names to be included in the final answer. It should be noted that many Mexican species are not registered in such resources, especially endemic organisms.

[7] http://resolver.globalnames.org/data sources!

One drawback using GNRD service is that it experiences long network delays when many large documents are trying to be resolved. Furthermore, after a long delay it is possible to receive empty answers or error codes. Consequently, we have developed an efficient caller which first divides the texts in lots, and for each one, it sends a request to the resolver. Finally, the responses are merged to have one single index. The main advantage of this strategy is that if one request fails, only a part of the index will be lost while the other lots could have non-empty responses. Moreover, each request could be asynchronous using a task manager library.

The command to create the index of names of a file using Global Names is:

```
# python txtm.py GnIndexController --infile f.txt
```

7. Automatic Extraction of uses of Biodiversity in Mexico

In terminology extraction, some methods based in syntactic patterns in Spanish have been proposed to detect functional definitions into specialized texts [9]. A functional definition is when a term *T* is associated with some specific use *U* by a syntactic pattern called functional verbal predication. This verbal predication is simply a verbal form generally used to describe *T*'s uses.

In this first version of the library, we have developed an extractor that identifies fragments about the use of Mexican species with the help of patterns of type "*T* + {fused as} + U". Example 2 presents the term *Tunillo*, the common name of a Mexican cactus; and the functional verbal predication *used as*. It is important to note that these patterns should be detected simultaneously in the same fragment and this is the reason why the segmentation by sentences presented in Section 4 is necessary.

In addition to common names, scientific names and verbal functional predications, there are other elements of interest that have been integrated such as parts of animals and plants, names in native languages, names of objects for domestic use, hunting and fishing instruments, names of conditions in indigenous

communities, and names of therapeutic practices, among others. All these resources were provided by experts at CONABIO.

> Example 2. Tunillo (Stenocereus treleasei):
> *se come y se vende el fruto, hay quien hace juguetes con los tallos, y se pega una parte del tallo detrás de las orejas cuando hay paperas, se usa también como cerco vivo.*[8]

Syntactic based methods, however, present two major disadvantages: (1) they may extract false positives, and (2) they cannot extract fragments that do not contain verbal patterns. Consequently, we plan to annotate and validate manually the fragments extracted on this stage in order to generate a dataset to train supervised learners that can extract this information from fragments that do not present all the syntactic elements.

We can extract this sort of fragments with our library using the following command:

```
  # python txtm.py FragmentController --infile f.txt \
      --regex_dir <path_to_regexp_dir>

> {"documentName": "f.txt",
    "lineNumber": 1,
    "documentFragment": "Tunillo ...",
    "commonName": [{"instance": "Tunillo","offsetStart": 1,
      "offsetEnd": 8},...],
    "sciName": [{"instance": "Stenocereus treleasei", "offsetStart": 10
      "offsetEnd": 31},...],
    "parts": [{"instance": "fruto","offsetStart": 55,
      "offsetEnd": 60},
      ...],
    "functionalVerb": [{"instance": "usa","offsetStart": 176,
      "offsetEnd": 179}],
    "handcraft": [{"instance": "juguetes","offsetStart": 77,
      "offsetEnd": 85},...]
}
```

[8] An approximated translation will be: Tunillo (Stenocereus treleasei): the fruit is eatable and sold; some others make toys using its stems, or stick a piece of its stalk behind ears when they have mumps. It is also used as a hedge.

8. PRELIMINARY RESULTS AND DISCUSSION

8.1. *Sentence segmentation*

In Table 1, we show the improvement in sentence frontiers detection after using specific domain corpora as well as a specialized dictionary of abbreviations to train a maximum entropy model. The best model was trained with 4.4G words (185.1M sentences) from the Environment and Medicine documents of the CTIULA[9] Technical Corpus [10] using 10,000 iterations and cuto equals to 4 as parameters.

Sentence frontiers detection is crucial to the rest of the tasks because many of them depend on the quality of text segmentation, as we mentioned above. For instance, in species names detection, it could be an important feature the fact that other taxon names appear in the same sentence.

Table 1. *Comparison of trained models for sentence frontiers detection*

	Precision	Recall	F-measure
General Spanish	0.6241	0.7011	0.6604
CT-IULA (environment & medicine)	0.8333	0.8897	0.8606
CT-IULA + ad hoc Abbreviations	0.9211	0.9003	0.9106

8.2. *Taxon names detection*

A crucial task for text mining for biodiversity literature is to find scientific names of species. In section 5, we have described an experimental scientific names indexer based on regular expressions (RegExp) containing around 1000 genera and 2600 species names of Mexican trees. Later in section 6, we have described Taxon Finder [3] and NetiNeti [5]. The former based on lexicons and the later based on machine learning methods.

Table 2 presents the results for the evaluation of different methods for scientific names detection: a tree specialized Regex, Taxon Finder and a NetiNeti model trained for biodiversity literature in Spanish. For this last, we created a dataset to train NetiNeti models in Spanish, the best NetiNeti model obtained

[9] http://www.iula.upf.edu/corpus/corpusuk.htm

with literature in Spanish is what we call SpaNeti.We also show the results using the default train parameters for English NetiNeti to point out the improvement after using texts in Spanish to train the model. We have evaluated all of the methods using the same, manually annotated, text about Mexican trees [11]. Evaluation is composed of three sub-tasks: to seek out one-word taxons (Monomial), to seek out two-word taxons (Binomial) and to seek out any length names (Any Taxon).

Table 2. *Comparison of tools for taxon names detection in a text about trees*

	Monomial		Binomial		Any Taxon	
	Precision	Recall	Precision	Recall	Precision	Recall
RegExp (ad hoc trees)	1.0000	0.6821	1.0000	0.5748	0.9117	0.4033
Taxon Finder	0.8754	0.9014	0.9352	0.8019	0.8339	0.8587
SpaNeti (Spanish model)	0.9120	0.6171	0.9545	0.7608	0.8379	0.5669
NetiNeti (Default model)	0.4383	0.6542	0.6494	0.7874	0.3872	0.5780

As can be expected, extracting names based on RegExp is limited to the dictionary employed which is relfected on the low recall. TaxonFinder presents a more stable result than any other method used but there is an improvement on the extraction if we use SpaNeti. Therefore we propose that the best strategy for scientific names detection is to combine both methods.

Below we are going to discuss more thoroughly what are the advantages and disadvantages of each method.

Regex Using regular expressions we obtain a perfect precision score of 1.0 for Monomial and Binomial names tasks but not for Any Taxon because in this last task we have considered complete names as the correct answer. In consequence, the name “Pinus pseudostrobus” found by Regex is penalized against the more specific, say longer name “Pinus pseudostrobus var. oaxacana”. Indeed, our experimental Regex had not rules about names whose length is greater than two words. After our experiments, we found out that trying to capture all possible taxon name cases in one single expression could be very challenging. Hence, not the easiest strategy.

The second observation is that although the test text is about trees, it is common to find names of species other than trees. Therefore, the recall is penalized.

However, the most significant drawback of Regex is that it does not recognize subtle differences in species names. It does not match taxon names if they are not written exactly as they appear in the regular expression. For instance, if an author has named "magnolifolia" (instead of "magnoliifolia") the taxon will not be retrieved. In example 3 "Q. glaucoides" and "Quercus" are detected but "Q. magnolifolia" is not.

> *Example 3. La especie con los valores más altos de importancia es Q. magnolifolia (185.29) esta especie presenta los valores más altos de densidad (60), frecuencia (50) y cobertura (75), los otros encinos presentes en este sitio presentan valores bajos de importancia Q. glaucoides (26) Quercus 346 (24) y Quercus 347 (20), sin embargo es el único sitio de la zona intermedia con más de una especie de encinos, también es el unico sitio en el que se encontró al Timbre con valores de importancia moderados (44.27) así como de frecuencia (20) y cobertura (11) (Cuadro 7) (Figura 12).*

Taxon Finder This method obtained the best scores (over 80%) of recall for all tasks. The advantage of Taxon Finder is that it uses dictionaries from more than two levels in the taxonomic hierarchy, and rules to detect inferior levels like subspecies, race and variety. This enables it to identify long names like "Pinus pseudostrobus var. oaxacana". Taxon Finder, however, like the regex method does not recognize names with subtle differences in writing. In consequence, the same omission in Example 3 is expected.

An interesting aspect of results from Taxon Finder is that sometimes, names of rivers or locations are confused with species. In example 4, "Atoyac" and "Bejuco" are retrieved because the former is included in the genera dictionary and the second is confused with "Bejuco pendulus Loe." (a synonym of "Hippocratea volubilis L."). Similar phenomena occurs for "Calera", "Jarilla" and "Huerta".

Example 4. La zona forma parte de la región hidrológica Río Atoyac (RH-20), destaca el Río Molino que nace del Río la Catrina, Río Obscuro y Río Bejuco a una altitud de 2900 msnm en las faldas de la ladera sur de la Peña Boluda o Peña de San Felipe, que por su caudal es la principal corriente que llega a la población (INEGI, 2006)

SpaNeti To get more flexibility in the detection of scientific names we can use machine learning techniques, NetiNeti is a tool designed for this purpose, but is trained for English. SpaNeti, our best model trained with literature in Spanish obtained the following results.

SpaNeti gives many false positives with words starting with uppercase and ending in {“a”, “s”} like the following examples, some of them proper nouns: “Resulta”, “Catarina”, “Primaria”, “Secundaria”, “Frecuencia”, “Cobertura”, “Toda”, “Piedra”, “Esta”, “Oaxaca”, “Biznaga”, “Pingüïca”, “Bretónica”, “Mata”, “Higuerilla”, “Jarilla”, “Salvia”, “Ambas”, “Naturales”, “Medicinales”, “Algunas”, “Ornamentales”, “Centrales”, “Comunales”.

Since Spanish is a Romance Language it is natural that some heuristics that work to distinguish scientific names in texts in English do not work for texts in Spanish. We believe that the problem described above is due to some rules that NetiNeti uses for the ending letters that increase the probabilities for words ending in {“a”, “I”, “s”, “m”}.

But the strength of SpaNeti is that it is capable of finding names that do not have the orthography of the lexicon, these are some examples that SpaNeti does detect: “Coryphanta retusa” (Coryphantha retusa), “Quercus magnolifolia” (Quercus magnoliifolia), “Abies hickeli” (Abies hickelii).

9. Conclusions and Future Work

We have presented in this article the first version of a long term project that aims to develop tools for extracting information from biodiversity literature in Spanish. The need for this system was identified from the lack of text mining tools for Life Science

literature in Spanish. We have identified some important tasks that need to be completed in order to start applying formal data mining techniques for this purpose, and this first delivery of our Text Mining Library for Biodiversity Literature in Spanish is an effort to compile a set of tools that facilitate the preprocesing tasks that many text mining projects will require before starting to apply NLP techniques. We have also included some basic capabilities that may help finding information about biodiversity, in particular about Mexican trees.

In this first stage we have dealt with extraction of text from PDF files, correction of OCR, taxon recognition and extraction of fragments that are likely to have information about traditional uses of Mexican species. We have shown the advantages of training a model to correct sentences in the specific domain of biodiversity and we have discussed our results in Named Entities Recognition focused on species names.

There are many improvements and lines of work required for this library. To name a few, we need a better model to correct sentences in this specific domain, also entities recognition focused on species names and common names disambiguation is a common task that would be very useful to several projects. As part of the current development, we have been compiling a dataset to apply machine learning techniques. And in future we plan to include the annotation of the corpus to start experiments in other complex tasks like automatic detection of the relationship between a taxon and its aliases. We plan to publish this dataset so that it can serve as a benchmark to other NLP projects in Spanish.

There are many things that need to be done, our hope is that this work will generate interest in the community to contribute in this project full of potential applications.

Figure 1 shows part of the library architecture. In the current version, the controllers layer (in the middle of the figure) enables the access to lower level functionality classes (at the bottom of the figure). The natural way to add functionality to the library is by creating a new branch in the repository and then adding the

unit tests[10] for the new controller (following the test driven development practices mentioned in [12]).

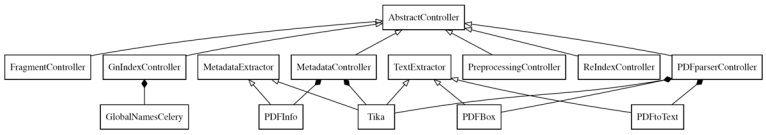

Figure 1. *UML class diagram*

REFERENCES

1. Freeland, C. 2011. Digitization and enhancement of biodiversity literature through ocr, scientific names mapping and crowd sourcing. *BioSystematics Berlin*.
2. Gerner, M., Nenadic, G. & Bergman, C. M. 2010. Linnaeus: A species name identification system for biomedical literature. *BMC bioinformatics*, 11, 85.
3. Leary, P. R., Remsen, D. P., Norton, C. N., Patterson, D. J. & Sarkar, I. N. 2007. ubiorss: tracking taxonomic literature using rss. *Bioinformatics*, 23, 1434-1436.
4. Koning, D., Sarkar, I. N. & Moritz, T. 2005. Taxongrab: Extracting taxonomic names from text. *Biodiversity Informatics*, 2, 79-82.
5. Akella, L. M., Norton, C. N. & Miller, H. 2012. Netineti: Discovery of scientific names from text using machine learning methods. *BMC Bioinformatics*, 13, 211.
6. Tang, X. & Heidorn, P. 2007. Using automatically extracted information in species page retrieval. *Proceedings of TDWG 2007*.
7. Krauthammer, M., Rzhetsky, A., Morozov, P. & Friedman, C. 2000. Using blast for identifying gene and protein names in journal articles. *Gene*, 259, 245-252.
8. Ride, W. D. 1999. International code of zoological nomenclature. *International Trust for Zoological Nomenclatu History Museum*.
9. Sierra, G.E., Alarcón, R., Molina, A. & Aldana, E. 2009. Web explotation for definition extraction. In *IEEE Latin AmericanWeb Congress (LA-WEB'09)* (pp. 217-223), Merida, Mexico.
10. Vivaldi Palatresi, J. 2009. Corpus and exploitation tool: Iulact and bwananet. In *International Conference on Corpus Linguistics*

[10] Test data are provided in the repository.

(CICL 2009) (pp. 224-239), *A survey on corpus-based research*, Universidad de Murcia.
11. Padilla Gómez, E. 2007. Estudio ecológico y etnobotánico de la vegetación del municipio de san pablo etla, oaxaca. Master's thesis, Centro Interdisciplinario de Investigación para el Desarrollo Integral Regional, Unidad Oaxaca. Instituto Politénico Nacional.
12. Astels, D. 2003. Test driven development: A practical guide. Prentice Hall Professional Technical Reference.

JUAN M. BARRIOS
THE NATIONAL COMMISSION FOR KNOWLEDGE AND USE OF BIODIVERSITY (CONABIO)
E-MAIL: <AMOLINA@CONABIO.GOB.MX>

ALEJANDRO MOLINA
THE NATIONAL COMMISSION FOR KNOWLEDGE AND USE OF BIODIVERSITY (CONABIO)

RAUL SIERRA-ALCOCER
THE NATIONAL COMMISSION FOR KNOWLEDGE AND USE OF BIODIVERSITY (CONABIO)

ENRIQUE-DANIEL
THE NATIONAL COMMISSION FOR KNOWLEDGE AND USE OF BIODIVERSITY (CONABIO)

ZENTENO-JIMENEZ
THE NATIONAL COMMISSION FOR KNOWLEDGE AND USE OF BIODIVERSITY (CONABIO)