ANTEATER

Report

© http://en.wikipedia.org/wiki/User:Brian.gratwicke

Julia Damerow
September 11, 2012

ANTEATER

Report

Julia Damerow
September 11, 2012

Project Goal

The goal of this project is the extraction the information listed below from texts downloaded from the Federal Register (https://www.federalregister.gov). The texts are mainly applications for permits, notices about given permits, etc.

- Basic information to be extracted:
- Applicant of permit (if applicable, institution applicant is working for)
- Location of applicant
- Researched species (by scientific name)
- Location of research

If there is enough time, the following information are desired in addition to these basic information:

- Researched species (by common name)
- Purpose of research

What action was applied to permit (requested, issued, renewed, amended or denied)

Concept

To accomplish the described task a software component called “Anteater” (*an*notation *t*ool to extract *e*ndangered *a*nimals from *te*xt *r*esources[1]) is being developed. The basic idea of Anteater is the following:

Extract names of people (and if possible institutions), species names and places

Filter extracted information to retrieve applicants, species and location of applicants/locations of research

Associate people with species and locations according to “who research on what species where.”

[1] Sorry, if this is not entirely correct, but it works so well as acronym ☺

Implementation

This section will give a brief summary on each of the three steps listed above. In step one (extraction of information from text), candidates are generated that are filtered in step two. Step 3 filters out even more candidates if certain requirements are not met (see below).

After examining texts that were given as samples, it became clear that almost all relevant information is contained in two parts of the documents: summary and supplementary information. Therefore, the text used for extraction of information is reduced to these two sections in the beginning.

Step 1: Extract people, species and places

Each type of information (people, species and place) is extracted differently. Names of peoples are extracted using the Stanford NLP library (http://nlp.stanford.edu/). This library has a named entity recognizer (NER) that is employed to generate candidates for applicant names. The NER returns a text marked up with persons, locations and organizations.

To extract species the text is send to the Global Names Recognition and Discovery (GNRD) webservice (http://gnrd.globalnames.org/). GNRD returns XML that is parsed to generate candidate species.

The extraction of places works similar to the extraction of species. Only difference is that the text is send to Yahoo! Placemaker (http://developer.yahoo.com/geo/placemaker/).

Step 2: Filter candidates

The results created in step one have different qualities. Therefore, for each kind of information a different filter mechanism is used. The GNRD service has a very high precision rate; almost all found species names are actually species names. To find the few names that are incorrectly found, names are checked against found places and filtered out if Placemaker found them, as well.

To filter people found by the Stanford NLP library, a machine-learning algorithm is used. Its input are all persons, locations and organizations found by the Stanford NER since it is not very accurate in classifying (e.g. a organization might be tagged as location). Each instance in the model (each candidate for being an applicant) analyzed by the algorithm is described by 20 features, such as length of a name (person name, institution name, etc.), is it subject of the sentence, etc. (see appendix A for a list of all features). The model was trained on 186 instances using a “LADTree” classifier. With a training set of 2/3 of all instances and a test set of 1/3 of the instances, the results were 90.5% correctly classified instances and 9.5% incorrectly classified instances. The model used in Anteater was trained with the complete training set using cross-validation.

To filter location results a machine-learning algorithm is used as well. The algorithm used is a “LMT” tree classifier. The classifier was trained on 266 instances with each 20 features (see appendix A). It tries to classify four classes: not relevant, location of research, location of applicant and institution of applicants. Training results are 86.5% correctly classified instances and 13.5% incorrectly classified instances.

Step 3: Creating "research events"

The results of step 2 are used to create so-called "research events." A research event contains applicants of a permit/application, their locations and if applicable and available their institutions, species that the applicant applied to conduct research on, and the locations research would be conducted.

To create these events, several rules where defined to collect and sort the information found in step 2. A list of rules can be found in Appendix B.

Results

The quality of the end results (the research events) is highly dependent on the candidates generated in step 1. If an applicant, location or species is not a candidate, it can't be recognized as part of an event. Fig. 1 shows for a test set of 13 texts how much correct information was generated for all events of a text.

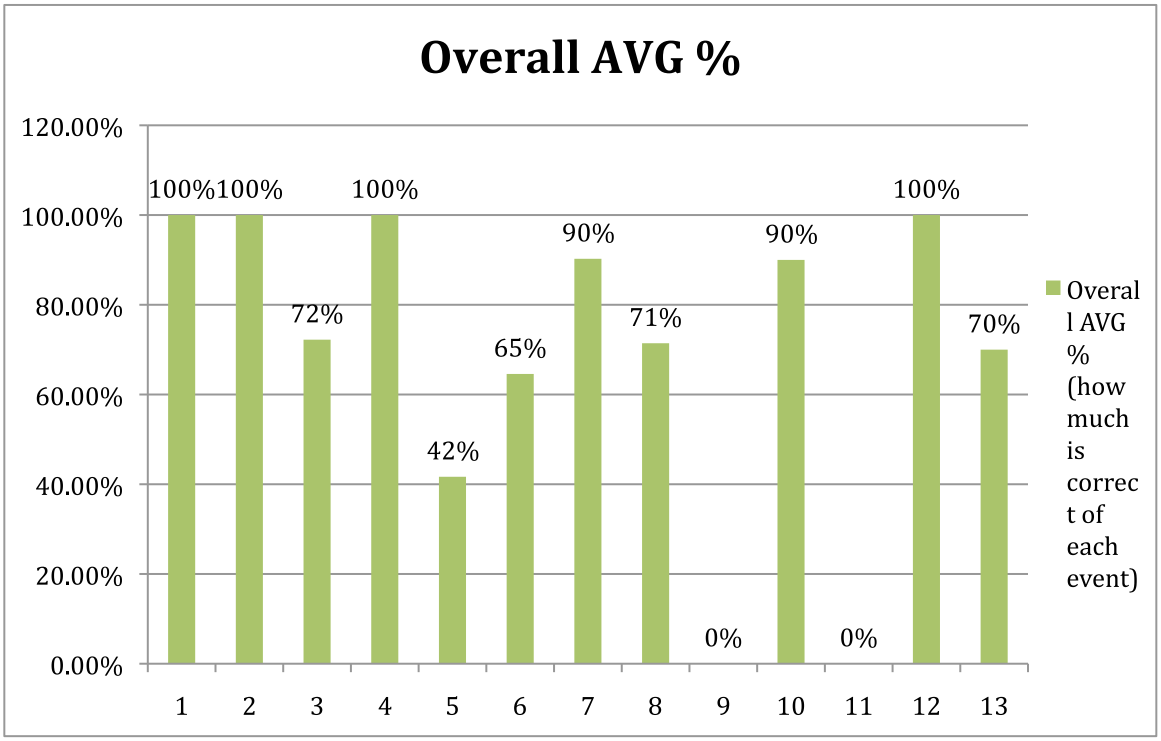

Fig. 1: Completeness of research events generated for 13 texts

In general, the more standardized a text is, the better the results. For example, text number 12 has the following summary:

> “Notice is hereby given that Lawrence D. Wood, Marinelife Center of Juno Beach, 14200 U.S. Hwy. #1, Juno Beach, FL, 33408, has been issued a permit to take hawksbill sea turtles (Eretmochelys imbricata) for purposes of scientific research.”

This summary contains all information in a standardized form and can therefore be easily retrieved. The same is true for texts that contain several permits/applications and each is listed in the supplementary information section starting with “Applicant: “ (for instance text 7).

Anteater doesn’t perform too well in cases where several applicants are listed in the summary and are then referred to by permit number in the supplementary information section. Especially since there are often spelling mistakes in the permit numbers; e.g. in the summary the permit numbers is 715-1614-00:

> “NMFS has received a permit application from: Ocean Alliance/Whale Conservation Institute, 191 Weston Road, Lincoln, Massachusetts 01773 (Dr. Roger S. Payne, Principal Investigator) (Application No. 751-1614-00);[...]”

But in the supplementary information section the, permit is referenced with 715-1614-00:

> “For Application No. 715-1614-00, the applicant requests permission to conduct vessel and aerial surveys, [...]”

Output

Anteater creates the following files (all in XML format):

Analysis files — These files contain the results from step 1 and the text that is analyzed.

Machine Learning files — These files contain the data sets that are used in the machine learning component in step 2 (for applicants and locations).

Pre-result files — These files contain the analyzed texts (summary and supplementary information) marked up with found applicants, locations and species.

Event files — These files contain the generated research events and are the final results.

How to use Anteater

Anteater is distributed as a jar-file. You run it form the command line like that:

```
java -Xmx2g -jar anteater.jar T A ML R E
```

where

```
T = absolute path to texts folder

A = absolute path to where analysis files should be stored

ML = absolute path to where machine learning files should be stored

R = absolute path to where result files should be stored

E = absolute path to where event files should be stored
```

Future work

There are two areas of future work:

1. further develop Anteater so that it finds additional information as described in the project goal section above;
2. improve Anteater in extracting basic information.

There are several ways Anteater could be improved:

a. Create more training data for the machine learning algorithms that filter applicant candidates and locations.

b. Develop additional features for both machine learning algorithms.

c. Try to find a different library to analyze sentences (e.g. find subject) than the Stanford NLP library to increase Anteater's speed.

d. Develop machine learning component to put together research events instead of the current rule-based component.

e. Try additional or different way to generate candidates for applicants, locations and species.

Appendix A – Machine Learning Model Features

Machine learning features to filter applicants:

Name of feature	Type	Description
isApplicant	Integer (0=false, 1=true, ?=unknown)	This features is set in training data and has "?" in real data where we need a prediction.
text_type	Integer (1=summary, 2=supplementary information)	Is candidate appearing in a summary or supplementary information section?
name_length	Integer	Length of candidate.
issued	Integer (0=false, 1=true)	Does sentence of candidate contain the word "issued"?
applied	Integer (0=false, 1=true)	Does sentence of candidate contain the word "applied"?
permit	Integer (0=false, 1=true)	Does sentence of candidate contain the word "permit"?
comment	Integer (0=false, 1=true)	Does sentence of candidate contain the word "comment"?
is_subject	Integer (0=false, 1=true)	Is candidate the subject of a sentence?
applicant	Integer (0=false, 1=true)	Does sentence of candidate contain the word "applicant"?
char_applicant_to_name	Integer (positive if "applicant" is after candidate, negative if before)	How many characters are between the end of word "applicant" and the beginning of candidate in paragraph of candidate? If the term "applicant" appears several times, take smallest number.
pers_org_loc	Integer (1=person, 2=organization, 3= location)	Is candidate classified by NER as person, organization or location?

GNRD-nlp_overlap_nlp	Float	If candidate was also found by GNRD, how many characters are similar to GNRD results? Calculate: Common characters of both results/# characters of candidate
GNRD-nlp_overlap_GNRD	Float	If candidate was also found by GNRD, how many characters of GNRD result are similar to candiddate? Calculate: Common characters of both results/# characters of GNRD result
start_idx_equals_GNRD	Integer (0=false, 1=true)	If candidate was also found by GNRD, does start index equals start index of GNRD result?
Placemaker-nlp_overlap_nlp	Float	If candidate was also found by Placemaker, how many characters are similar to Placemaker result? Calculate: Common characters of both results/# characters of candidate
Placemaker-nlp_overlap_pl	Float	If candidate was also found by Placemaker, how many characters of Placemaker result are similar to candiddate? Calculate: Common characters of both results/# characters of Placemaker result
start_idx_equals_placemaker	Integer (0=false, 1=true)	If candidate was also found by Placemaker, does start index equals start index of Placemaker result?

surrounded_by_brackets	Integer (0=false, 1=true)	Is candidate surrounded by brackets?
surrounded_by_commata	Integer (0=false, 1=true)	Is candidate surrounded by commata?
followed_by_s	Integer (0=false, 1=true)	Is candidate followed by “’s”?
isAbbreviation	Integer (0=false, 1=true)	Is candidate an abbreviation according to Stanford dependency parser?

Machine learning features to filter locations:

Name of feature	Type	Description
location_type	Integer (0=other, 1=research location, 2=applicant location, 3=applicant institution)	This features is set in training data and has “?” in real data where we need a prediction.
numbers/words	Float	Of all words contained in candidate, how many are numbers? Calculated by: # numbers / # all words
starts_with_uppercase/words	Float	Of all words contained in candidate, how many start with an uppercase letter? Calculate by: # words starting with uppercase letter / # all words
contains_2_uppercase_letter_word	Integer (0=false, 1=true)	Does candidate contain a word consisting of two uppercase letters (e.g. AZ)?
contains_university	Integer (0=false, 1=true)	Does candidate contain the word “university?”
surrounded_by_commata	Integer (0=false, 1=true)	Is candidate surrounded by commata?

surrounded_by_brackets	Integer (0=false, 1=true)	Is candidate surrounded by brackets?
preceeded_by_and	Integer (0=false, 1=true)	Is candidate preceded by the word “and?”
preceeded_by_the	Integer (0=false, 1=true)	Is candidate preceded by the word “the?”
char_to_last_species_in_p	Integer	How many characters are between beginning of candidate and end of preceding species (in same paragraph)?
char_to_last_species_in_p	Integer	How many characters are between end of candidate and beginning of next species (in same paragraph)?
char_to_study_in_p	Integer (positive if “study” is after candidate, negative if before)	How many characters are between the beginning of word “study” and the beginning of candidate in paragraph of candidate? If the term “study” appears several times, take smallest number.
char_to_studies_in_p	Integer (positive if “studies” is after candidate, negative if before)	How many characters are between the beginning of word “studies” and the beginning of candidate in paragraph of candidate? If the term “study” appears several times, take smallest number.
char_to_in_in_s	Integer (positive if “in” is after candidate, negative if before)	How many characters are between the beginning of word “in” and the beginning of candidate in paragraph of candidate? If the term “study” appears several times, take smallest number.
char_to_at_in_s	Integer (positive if “at” is after candidate, negative if before)	How many characters are between the beginning of word “at” and the beginning of candidate in paragraph of candidate? If the term “study” appears several times, take smallest number.

nr_char_to_last_applicant_in_text	Integer	How many characters are between beginning of candidate and end of last applicant before candidate?
has_comma	Integer (0=false, 1=true)	Does candidate contain “,”?
has_brackets	Integer (0=false, 1=true)	Does candidate contains “(“ and/or “)”?
type	Integer (0=Other, 1=Town, 2=County, 3=State, 4=Country, 5=Suburb, 6=POI, 7=Zip, 8=Ocean)	What type has candidate according to Placemaker?
chars_to_survey_in_s	Integer	How many characters are between the beginning of word “survey” and the beginning of candidate in sentence of candidate? If the term “survey” appears several times, take smallest number.
chars_to_species_in_s	Integer	How many characters are between the beginning of word “species” and the beginning of candidate in sentence of candidate? If the term “species” appears several times, take smallest number.

Appendix B – Rules for Creating Research Events

1. If there is only one applicant in a text, use all information available to create research event.

2. If there are no applicants in summary but in supplementary information section, start new research event with each applicant unless there is no researched species in between two applicants. In that case put all applicants to next species in one event. Always use next applicant location for applicants.

3. If there are several applicants in summary, forget about applicants in supplementary information section. Start new research event with every applicant unless there is no application or permit number between two applicants. In that case put all applicants to next application/permit number in one event. Always use next applicant location for applicants. Store permit/application number to find corresponding paragraphs in supplementary information section to fill research event with research location and species.

Appendix C – Example Output

2010-11336.xml

Pre-result file (annotated text):

```
<?xml version="1.0" encoding="UTF-8"?>
<results>
    <summaries>
        <summary>
            <p type="2">SUMMARY:</p>
            <p type="1">
                Notice is hereby given that
                <applicant>NMFS Southwest Fisheries Science Center</applicant>
                (SWFSC) [Responsible Party:
                <applicant>Lisa Ballance</applicant>
                ], 3333 N. Torrey Pines Ct.,
                <applicant_location woeId="2434241" type="Suburb" name="La Jolla, San Diego, CA, US">La Jolla,
CA</applicant_location>
                92037, has requested a modification to scientific research Permit No. 1596-02.
            </p>
        </summary>
    </summaries>
    <supplementary_information>

        <supplInfo>
            <p type="2">SUPPLEMENTARY INFORMATION:</p>
            <p type="1">The subject modification to Permit No. 1596-02, issued on
                July 29, 2009 (74 FR 38585), is requested under the authority of the
                Endangered Species Act of 1973, as amended (16 U.S.C. 1531 et seq. )
                and the regulations governing the taking, importing, and exporting
                of endangered and threatened species (50 CFR 222-226).</p>
            <p type="1">
                Permit No. 1596-02 authorizes the SWFSC to capture, measure, weigh,
                blood and tissue sample, photograph, flipper and PIT tag, fat
                biopsy, ultrasound, satellite tag, and attach a VHF/TDR/sonic
                tag/video system, VHR/TDR/sonic tag/GPS unit, or VHR/TDR/sonic
                tag/GPS/video camera system to leatherback (
               <species_scientific name="Dermochelys coriacea">Dermochelys coriacea</species_scientific>
                ) sea turtles during research activities conducted off the western
                coast of the continental United States. Animals with the video
                camera system may be re-approached to collect the unit and then
                sampled, tagged, and have another video camera unit attached. The
                SWFSC requests authorization to use a direct tag attachment method
                in place of previously authorized harness attachments. These tags
                would provide valuable information on leatherback movements and
                behavior in the
                <research_location woeId="55959717" type="Ocean" name="Pacific Ocean">Pacific
Ocean</research_location>
                between their foraging areas and nesting beaches. No increase in the
                number of animals taken is requested. The research would continue to
                occur in waters off the coast of the
                <research_location woeId="23689947" type="Colloquial" name="US Western States, US">western
United States</research_location>
                through February 1, 2012.
            </p>
        </supplInfo>
    </supplementary_information>
</results>
```

Events-file:

```
<?xml version="1.0" encoding="UTF-8"?>
<events>
    <event text="2010-11336.xml" date_filed="5-11-10">
        <applicants>
            <applicant>
                <name>NMFS Southwest Fisheries Science Center</name>
                <applicant_institutions />
                <applicant_locations>
                    <applicant_locations>
                        <name>La Jolla, CA</name>
                        <place_information type="Suburb" woeId="2434241" latitude="32.8426" longitude="-
117.272">La Jolla, San Diego, CA, US</place_information>
                    </applicant_locations>
                </applicant_locations>
            </applicant>
            <applicant>
                <name>Lisa Ballance</name>
                <applicant_institutions />
                    <applicant_locations>
                    <applicant_locations>
                        <name>La Jolla, CA</name>
                        <place_information type="Suburb" woeId="2434241" latitude="32.8426" longitude="-117.272">La Jolla, San Diego, CA, US</place_information>
                    </applicant_locations>
                </applicant_locations>
            </applicant>
        </applicants>
        <research_locations>
            <research_location>
                <name>Pacific Ocean</name>
                <place_information type="Ocean" woeId="55959717" latitude="0.89316" longitude="-154.721">Pacific Ocean</place_information>
            </research_location>
            <research_location>
                <name>western United States</name>
                <place_information type="Colloquial" woeId="23689947" latitude="37.7138" longitude="-106.419">US Western States, US</place_information>
            </research_location>
        </research_locations>
        <researched_species>
            <species identified_name="Dermochelys coriacea">Dermochelys coriacea</species>
        </researched_species>
    </event>
</events>
```