Laboratory Project of ANLP-MAI for the course 2014-2015.

The Laboratory Project of ANLP-MAI for this course consists of building a geodisambiguation system.

The task of geodisambiguation consists on assigning to a Named Entity (NE), previously classified as a Location (or Geopolitical Entity, GPE), its correct referent (from a set of candidates). The task is sometimes named georeferencing or grounding, although often these terms imply providing the coordinates (latitude and longitude) of the NE. When the task consists on assigning to the location an entry in a reference Knowledge Base (KB, for instance, Wikipedia, WP, Freebase, or DBpedia, DBP) it is named as Entity Linking (EL).

Consider the following text, selected from the EL task in the TAC-KBP 2014 contest. The task consisted on linking to a KB a set of about 5,000 queries (from which 1/3 were GPE). Each query consisted of a name, a reference document and the offset of the query name in the reference document. The example included here correspond to the query “Fremont” in bold in the text:

Toya says it will pull out of a California factory venture that it previously operated jointly with General Motors and move production to other plants in the U.S., Canada and Japan. Toyota Motor Corp. said Friday that production of the Corolla subcompact will be moved to its Cambridge, Ontario, Canada plant, as well as to Japan. It says Tacoma pickups, another model made in the Fremont, California-based New United Motor Manufacturing Inc., or NUMMI, will be produced at its plant in San Antonio, Texas. But Toyota says it remains committed to the North American market, but has decided to end its contract with NUMMI and stop production there in March 2010. General Motors already has pulled out of the venture.

The text has been segmented into sentences and a NERC has been run on it. resulting in the following:

<ORGToyota> says it will pull out of a <LOCCalifornia> factory venture that it previously operated jointly with <ORGGeneral Motors> and move production to other plants in the <LOCU.S.>, <LOCCanada> and <LOCJapan>.

<ORGToyota Motor Corp>. said Friday that production of the <MISCCorolla> subcompact will be moved to its <LOCCambridge>, <LOCOntario>, <LOCCanada> plant, as well as to <LOCJapan>.

It says <MISTacoma> pickups, another model made in the <LOCFremont>, <LOCCalifornia>-based <ORGNew United Motor Manufacturing Inc.>, or <ORGNUMMI>, will be produced at its plant in <LOCSan Antonio>, <LOCTexas>.

But <ORGToyota> says it remains committed to the North American market, but has decided to end its contract with <ORGNUMMI> and stop production there in March 2010. <ORGGeneral Motors> already has pulled out of the venture.
Looking for “Fremont” in WP we access a disambiguation page with the following options (a similar result could be obtained from DBP):

- [[Fremont, California]]
- [[Fremont (Amtrak station)]]
- [[Fremont (BART station)]]
- [[Fremont Central Park]]
- [[Fremont, Yolo County, California]]
- [[Fremont Landing, California]]
- [[Fremont County, Colorado]]
- [[Fremont County, Idaho]]
- [[Fremont Township, Lake County, Illinois]]
- [[Fremont Center, Illinois]]
- [[Fremont, Indiana]]
- [[Fremont, Iowa]]
- [[Fremont County, Iowa]]
- [[Fremont, Michigan]]
- [[Fremont Township, Isabella County, Michigan]]
- [[Fremont Township, Saginaw County, Michigan]]
- [[Fremont Township, Sanilac County, Michigan]]
- [[Fremont Township, Tuscola County, Michigan]]
- [[Fremont, Missouri]]
- [[Fremont Hills, Missouri]]
- [[Fremont, Nebraska]]
- [[Fremont, New Hampshire]]
- [[Fremont, Steuben County, New York]]
- [[Fremont, Sullivan County, New York]]
- [[Fremont, North Carolina]]
- [[Fremont, Ohio]]
- [[Fremont, Utah]]
- [[Fremont, Virginia]]
- [[Fremont, Seattle, Washington]]
- [[Fremont, Wisconsin]]
- [[Fremont, Clark County, Wisconsin]]
- [[Fremont, Waupaca County, Wisconsin]]
- [[Fremont County, Wyoming]]

(other options, corresponding to PER, ORG, or, MISC occur as well).

The task to be faced is the following:

Given a NE of type GPE (as “Fremont” in the example) and a text where a mention of the GPE occurs you have to identify the correct referent of the GPE. The identification could be a WP page or a DBP resource. The intuition is that other NE occurring close to the query GPE could contribute to disambiguate it. For instance, very close to the query mention there is a mention “California-based” that heavily reduces the set of candidates, other mentions (“U.S.”) are useless because all the candidates are in USA, others, finally, as “San Antonio”, “Texas”, or “Japan” are harmful for the task. For simplifying the task only the sentence where the GPE occurs and the 4 surrounding sentences (2 before and 2 after the GPE, if existing) will be used for disambiguating. Also for
simplifying the task you should consider as sources of evidence only the NEs of type LOC occurring in the text, discarding NEs of type PER, ORG or MISC.

You will be provided with a set of queries like the one presented above. You should pre-process the texts using Freeling (http://nlp.lsi.upc.edu/freeling/) for performing at least NERC. For getting the candidates you could use WP (using wikitools) or DBP (using sparql through the DBpedia end point), or both.

You should split the set of queries provided into two subsets, a development set (90% of the queries) and a test set (10%). You can use freely the development set while building your system, then you should apply your system to the test set and manually evaluate the results providing a global accuracy score.

Several scenarios can be considered. You are free for selecting the one you prefer (or you consider the most appropriate for the task):

1. Returning the most likely candidate.
2. Returning a scored list of candidates (sorted by score).
3. Returning the candidates with a score over an absolute threshold.
4. Returning the n best scored candidates (select n).
5. other, more complex, criteria.

Different experiments have to be carried out for different source scopes:

1. scope^0. Just the sentence containing the GPE to be disambiguated.
2. scope^1 = scope^0 plus the previous and following sentence (if existing).
3. scope^2 = scope^1 plus the previous and following sentence (if existing).

The approach I recommend is the following:

Given a scope^i, get all the mentions of GPE occurring in the scope. We denote this set as mentions^i. mentions^i = {m^i_1, m^i_2, ..., m^i_j, ..., m^i_M}. Where MI is the length of mentions^i. In our example we have:

- mentions^0 = {<LOC Fremont>, <LOC California>, <LOC San Antonio>, <LOC Texas}.
- mentions^1 = {<LOC Fremont>, <LOC California>, <LOC San Antonio>, <LOC Texas>, <LOC Cambridge>, <LOC Ontario>, <LOC Canada>, <LOC Japan}.
- mentions^2 = {<LOC Fremont>, <LOC California>, <LOC San Antonio>, <LOC Texas>, <LOC Cambridge>, <LOC Ontario>, <LOC Canada>, <LOC Japan}, <LOC California}, <LOC U.S.>, <LOC Canada}, <LOC Japan}.

Note that several mentions can be identical in wording.

For each mention m^i_j we get the set of entities candidate (using WP or DBP). We name these sets as entities^i_j, entities^i_j = {e^i_j,1, e^i_j,2, ..., e^i_j,k, ..., e^i_j,M_I}, where MIJ is the number of candidate entities for the mention m^i_j. We note entities^i the set of all entities of all mentions in mentions^i. For each of the entity candidates, e^i_j,k, i.e. the k-ary candidate of the j-ary mention of i scope, we obtain its geographic coordinates (latitude and longitude). We name it as coord(e^i_j,k), being points in the earth. We define a distance between geographic points in the earth (Euclidean distance, spherical distance, ...).
We define a configuration as a mapping from mentions \( i \) to entities \( i \), so that each mention \( m^i_j \) in mentions \( i \) is mapped into a unique candidate entity. We name configurations \( i \) the set of configurations for mentions \( i \), configurations \( i \) = \{ \( c^1_i \), \( c^2_i \), ..., \( c^i_j \), ..., \( c^i_M_i \) \}. So a configuration \( c^i_j \) is a set of pairs \(<m, e>\). A valid configuration is a configuration so that two identical mentions should be mapped to the same entity.

You have to define the density of a configuration \( c^i_j \). A possible definition can be the following:

\[
density(c^i_j) = 1 - \frac{2 \sum k \leq n \sum k \leq n \text{dist}(\text{coord}(e^i_k),\text{coord}(e^i_j))}{n^2}
\]

\( \text{where } c^i_j = \{<m^i_1,e^i_1>,...,<m^i_k,e^i_k>,...,<m^i_n,e^i_n>\} \)

That is, we compute the sum of distances from each geographic point in the configuration to each other and we normalize the count dividing it by the number of points squared.

The idea is that closed geographical points are likely to reinforce each others and so dense configurations are better.

Obviously a form of computing the best, \( c^{i*} \) (or the \( n \) best) configurations is by exhaustive searching.

\[
c^{i*} = \arg \max_{c^i_j} \density(c^i_j)
\]

As the cost of computing the density of a configuration is quadratic on the number of geographical points involved this approach can be costly in the case of many different configurations involving many candidate entities. So in some cases a greedy search could be more adequate.

Many greedy approaches can be designed. Just one of them could be the following:

1. Generate an initial configuration \( c \), randomly or including the most important entities for each mention. Importance can be defined, for instance as the most populated place (where population can be obtained from WP or DBP) or the most extensive area (where extension can be obtained from WP or DBP), or, simply considering that country > state > county > city.
2. The density of \( c \) is computed.
3. From \( c \) we perform a local search using a unique operator that consists of changing one of the mappings of the configuration into a new valid mapping whose density is at least the same of \( c \). The valid children of \( c \) are added to the search space.
4. Steps 2 and 3 are repeated until no valid moves are possible.
5. The result is a local maximum likely close to the global maximum.
A third approach could be an hybrid one that performs greedy search until a manageable size allow us to move to an exhaustive search.

The final point of this labo case is an analysis of the wrong results in the test set and a proposal to face them (just a proposal, nothing has to be implemented, but this task is important for the final grade).

*Barcelona, 9th February 2015*