Second NLU assignment

Giorgio Scarton 225606

April 26, 2021

Evaluation of Spacy NER in CoNLL 2003

The first step of the evaluation consist of the reconstruction of the original corpus from CoNLL 2003, to do so has been created a function called `conll_to_string(path)`:

- **Input:** the absolute or relative path of the source file as string
- **Output:** a string made of all tokens from CoNLL separated by a white space
- **Description:** splits text imported from the input file in lines, for each not empty line splits the content at white spaces and, if different from *-DOCSTART-*, stores the first item in the array in the output string followed by a white space. Out of the for loop removes the white space at the end of the string

Then SpaCy Tokenizer rules has to be change to adapt to the CoNLL tokens, although this can lead to suboptimal performances from several tests on CoNLL provided files it came out that editing the tokenizer is more computationally efficient than post-process the tokenized document and performances of the identification are similar. A *WhitespaceTokenizer* class, which only splits token at spaces, has been defined to replace the built-in tokenizer, in such a way it is not required post-process tokenization. Anyway another post-process is required to adapt SpaCy *entity types* to CoNLL ones, the mapping has been made in an arbitrary way considering both SpaCy entity definitions accessible via token property and CoNLL ones[1], follows the map:

SpaCy	**CoNLL**
PERSON	PER
FAC	LOC
GPE	LOC
EVENT	MISC
LAW	MISC
NORP	MISC
LANGUAGE	MISC
WORK_OF_ART	MISC
Others	O

To apply the map to the SpaCy document has been defined a function `remap(doc)` which takes as input the document and returns the same document with updated token's *entity types*.

Token level performance

To evaluate token level performance a modified version of provided `evaluate` function has been exploited, the input to the evaluation function `evaluate_token(refs,hyps)` is a list of list of tuple, to provide such kind of input two functions has been created:

- `get_list_from_doc(doc)`
  - Input: a SpaCy document
  - Output: a list of list of tuple
  - Description: for each token generates the tuple *(text,NER)*, adds this tuple to a list and then collects these lists in an outer list
- `get_list_from_conll(path)`

[1] Introduction to the CoNLL-2003 Shared Task:Language-Independent Named Entity Recognition - Erik F. Tjong Kim Sang, Fien De Meulder - 2003

– Input: the absolute or relative path of the source file as string
– Output: a list of list of tuple
– Description: see previous function