DataTXT at #Microposts2014 challenge

Ugo Scaiella
Michele Barbera
Stefano Parmesan
Spaziodati Srl
Trento, Italy
{surname}@spaziodati.eu

Gaetano Prestia
Net7 Srl
Pisa, Italy
prestia@netseven.it

Emilio Del Tessandoro
Mario Verì
Dipartimento di Informatica
University of Pisa, Italy
deltessa@di.unipi.it
veri@di.unipi.it

ABSTRACT

In this paper we describe the approach taken for the "Making Sense of Microposts challenge 2014" (#Microposts2014), where participants were asked to cross reference micro-posts extracted from Twitter with DBpedia URIs belonging to a given taxonomy.

For this task we deployed DATATXT[1] which is the evolution of TAGME[3], the state-of-the-art topic annotator for short texts and which has proven to be very effective and efficient in several challenging scenarios[2].

Keywords

topic annotator, entity extraction, datatxt

1. INTRODUCTION

The #Microposts2014 challenge[1] focuses on the task of annotating micro-posts with DBpedia entities belonging to a given taxonomy. With respect to traditional Information Retrieval tasks, such data poses new challenges in terms of the effectiveness and efficiency of the algorithms and applications because data is so short and noisy that it is difficult to mine significant statistics that are rather available when texts are long and well written. Additionally, participants have to deal with the issue of associating extracted entities with the provided taxonomy, that makes this challenge even harder.

For this challenge, we deployed DATATXT, an entity extraction system that is the evolution of TAGME[3]. An instance of DATATXT has been specifically trained using the official training set provided in this challenge.

This work has been supported by "'SenTaClAus"', a project funded by Tuscany Region (Italy).

2. ANATOMY OF DATATXT

DATATXT is able to identify meaningful sequences of one or more terms in unstructured texts on-the-fly and with high accuracy, and link them to a pertinent Wikipedia page. DATATXT maintains the core algorithm of its predecessor, TAGME, but adds functionality and several improvements in terms of cleaning the input text and identifying mentions.

The algorithm is based on the anchor texts drawn from Wikipedia for identifying mentions in input text. When an input text is received, it judiciously cross-references each anchor a found in the input text T with one pertinent page p_a of Wikipedia. DATATXT first identifies for each anchor a all possible pages p_a linked by a in Wikipedia. Then, from these pages, it selects the best association $a \mapsto p_a$ by computing a score based on a "collective agreement" between the page p_a and the pages p_b that can be associated with all other anchors $b^1 ... b^n$ detected in T. We deploy a voting-schema, where pages p_b vote for each candidate p_a according to a function that estimates the relatedness between two Wikipedia pages by exploiting the underlying graph. Further details of this voting-schema and the relatedness function can be found in [3]. Not all mentions extracted in this way are worth annotating, so a confidence score is assigned to all mentions. This score is based on (a) a–priori statistics based on Wikipedia and (b) other figures representing the coherence of the candidate entity with respect to the whole text. It is thus possible to discard those whose confidence score is below a given threshold.

DATATXT does not rely on any linguistic feature, but only on statistics and data extracted from Wikipedia. We argue that this approach, derived from TAGME, yields better results when dealing with user generated content such as micro-posts, where well-known NLP tools, such as part–of–speech taggers, are less effective because texts are short, fragmented and often contain slang and/or misspelled words. An in-depth evaluation of TAGME's effectiveness and comparison with others annotators was recently published in [2], showing the validity of this approach.

3. TRAINING

DATATXT was designed for short texts, but it is effective for long texts as well[2]. However there are some parameters that can be amended in order to better fit the context of this challenge. One of them is ϵ, which is used to tune the disambiguation algorithm[3] and defines whether DATATXT should rely more on the context or favor more common topics in order to discover entities. Using a higher value favors more common topics, which may lead to better results when processing fragmented inputs where the con-

[1] `http://dandelion.eu/datatxt`

text is not always reliable. Two other parameters have been taken into account: (a) the minimum link probability, say δ, that is used to discard a mention that is rarely used as anchor texts in Wikipedia; (b) the minimum commonness, say γ, that is used to discard a possible association $a \mapsto p_a$, thus reducing the "ambiguity" of a mention. Refer to [3], for further details on these two thresholds. DATATXT assigns a confidence score to each annotation so that those that are below a given threshold, say ϕ, can be discarded. This parameters can be used to balance precision vs. recall and the best value may vary based on the application context. For each configuration we tested, we evaluated the results using 20 values of this threshold, ranging from 0 to 1.

Another important issue we faced, is that the annotation task of this challenge has been restricted to entities belonging to a limited taxonomy. DATATXT is a generic *topic* annotator and it extracts all topics contained in the input text. If we considered the overall output produced by DATATXT, the results, and in particular the precision, would be significantly penalized because DATATXT also includes *topics* that are not part of the taxonomy. As an example consider this tweet, which was part of the training set: "*Bank of America posts $8.8 billion loss in second quarter due to mortgage security settlement*". The human annotators extracted *Bank of America* as the only mention of this micropost, whereas DATATXT extracts also *mortgage* and *security* linking them to `Mortgage_loan` and `Security_(finance)` respectively. These are not errors of the system, but #MSM2014 focused on a limited taxonomy and *mortgage* and *security* are not part of it. Unfortunately, given a DBpedia URI it was not possible to automatically check whether or not the entity belongs to that taxonomy. To address the issue, we initially tested a naive approach using a white–list of entities derived from the training set. This is useful but, of course, is not generic. Thus, we designed another approach that provides the probability that a generic entity belongs to the taxonomy based on the Wikipedia categories and DBpedia types associated with the entity. We thus gathered all Wikipedia categories and all DBpedia types associated with each entity extracted by DATATXT from the training set. We then counted the occurrences of categories and types for all entities that were part of the ground-truth and the occurrences of categories and types for those that were not. For each category/type we then computed a probability. Given an entity e extracted by DATATXT, we thus computed the probability that e belongs to the taxonomy by computing a weighted sum of probabilities of all categories and types of e. Finally, we discarded from the results all entities whose probability is below a given threshold. The value of this threshold, called β, was experimentally evaluated using the training set, together with other parameters mentioned above. We also tested a third approach by deploying a C4.5 classifier, which was trained by exploiting types and categories derived as mentioned above. Categories and types were thus deployed as features to train the classifier.

For parameters tuning, we simply used a grid search in an N–dimensional space, using a 5-fold cross evaluation, in order to avoid over–fitting. Note that DATATXT is very efficient, and the evaluation of a single parameter combination (ie the annotation of more than 2K tweets) takes about 800 ms, thus this search is feasible even with a long list of combinations. Tuned values of parameters do not change significantly across the different folds, showing a good stability and generality of the approach (see Table 1).

Fold#	β	γ	δ	ϵ	ϕ	F_1
1	0.4	0.4	0.3	0.15	0.7	0.5985
2	0.4	0.4	0.2	0.2	0.65	0.5853
3	0.3	0.5	0.2	0.2	0.7	0.5722
4	0.4	0.5	0.2	0.2	0.7	0.5690
5	0.4	0.4	0.2	0.2	0.6	0.5737

Table 1: Tuning second approach, results per single fold of cross evaluation

Approach	Precision	Recall	F_1
1. White–list only	66.3	41.3	50.2
2. White–list + types prob.	65.6	50.7	57.2
3. White–list + C4.5 classifier	75.8	55.5	64.1

Table 2: Results of our approaches.

4. RESULTS

During the training phase, we noticed several differences between the annotation generated by DATATXT and the annotation provided in the ground-truth, therefore we implemented a few post–annotation steps to improve the performance for this challenge: (a) DATATXT does not annotate dates or numbers, so a step that identifies these types of mentions using simple regular expressions was added; (b) DATATXT annotates only the first occurrence of a mention, so a post–processing step to handle repeated mentions was added. These steps do not affect the core algorithm and thus were not considered during the training phase, however they improve the performance of DATATXT for this challenge. Table 2 shows the overall results of the cross evaluation of our approaches using the training set. These figures are not directly comparable those presented in [2], as #MSM2014 focused on a limited set of entities, i.e. the ones specified by the taxonomy.

5. CONCLUSIONS

We have described the approach taken by our group for the #MSM2014 challenge, where we deployed DATATXT, the evolution of the state-of-the-art topic annotator TAGME. Given that its algorithm does not depend on linguistic features, DATATXT is very accurate even in this scenario. We have also outlined a basic approach to verticalize the general–purpose extraction algorithm to improve the performance in the domain defined within this challenge. We believe that this approach to verticalization could be further refined by applying more sophisticated machine–learning techniques, such as SVM or CRF.

6. REFERENCES

[1] A. E. Cano and others. #Microposts2014 NEEL Challenge. In *WWW 2014 Making Sense of Microposts Workshop*, 2014.

[2] M. Cornolti, P. Ferragina and M. Ciaramita. A framework for benchmarking entity-annotation systems. In *WWW*, 249–260, 2013.

[3] P. Ferragina and U. Scaiella. Fast and Accurate Annotation of Short Texts with Wikipedia Pages. *IEEE Software*, 29(1): 70–75, 2012.