A Strategy for the Integration of Named Entity Extraction and Linking Results

Jose L. Martinez-Rodriguez[1], Julio Hernandez[1], Ivan Lopez-Arevalo[1], and Ana B. Rios-Alvarado[2]

[1] Cinvestav-Tamaulipas
Victoria, Mexico
{lmartinez,nhernandez,ilopez}@tamps.cinvestav.mx
[2] Facultad de Ingeniería y Ciencias,
Universidad Autónoma de Tamaulipas,
Victoria, Mexico
arios@docentes.uat.edu.mx

Abstract. The extraction of named entities and their linking to a Semantic Web knowledge base is a task whose results support the process of extracting potentially useful elements of information from unstructured text. In the last decade, this task has been widely addressed with the aim of making possible the interconnection, exchange, and query of data on the Semantic Web. In this sense, several approaches based on the idea of ensemble methods (like those in Machine Learning) have been proposed to combine distinct named entity extraction and linking techniques in order to get better results than using a single such technique. Although the idea is to exploit features provided by diverse approaches to extract and link named entities from text, there are some issues to solve for integrating their results (e.g., heterogeneous output, duplicated entities). In this paper, we propose a strategy to integrate the output provided by some entity extraction and linking tools in an ensemble-like scheme. For such purpose, we consider steps for collecting and merging results supported by filtering decisions to overcome issues such as duplicated and/or overlapped entities. The results showed an increased performance in terms of the F-measure compared to isolated approaches.

Keywords: Named Entity Disambiguation, Entity Linking, Entity Joining, Ensemble Extractor

1 Introduction

The Semantic Web provides an extension of the Web in order to give a semantic and formal representation of data, in such a way that the information could be shared and reused by different applications [1]. In order to achieve this goal, several standards and protocols have been published on the Web, for example, the Resource Description Framework (RDF) and the Linked Open Data (LOD) principles [2]. The latter establishes that the information should be identified

using the Internationalized Resource Identifier (IRI) standard to ensure data interconnection and retrieval through the Internet.

On the other hand, in the field of text mining, a named entity (NE) is an important piece of information that refers to real or abstract things of the world, such as names of persons, places, dates, among others. In the context of the Semantic Web and following the LOD principles, a named entity has associated a unique IRI, which is used to describe it in a Knowledge Base (KB). In knowledge extraction, a common task is to look up for named entities and link them to a KB. This process is known as Entity Extraction and Linking (EEL, a.k.a, named entity disambiguation or entity linking) [10], where DBpedia[1] and YAGO[2] are the commonly used KBs. The result of this task is a tuple (¡EM, IRI¿) composed of a text fragment containing the entity mention (EM) and its identifier (IRI) from some KB. Different approaches for EEL have been proposed in the literature in order to cover varied types of entities [7,6,5,4]. Such approaches consider distinct domains, KBs, algorithms, and so on. In recent years, approaches such as [8,3] are based on the idea of ensemble systems (as presented in Machine Learning) for integrating several EEL tools. The aim is to exploit their features in such a way that a higher number of entities can be extracted without decreasing the system performance. Although such systems may increase the number of extractions, the integration of several EEL systems involves taking decisions such as duplicated and overlapped results. In other words, extracted entities may be contained within other extractions or duplicated with respect to the mention string (entity mention) but with a distinct identifier (IRI) or exactly equivalent (mention and IRI matching).

This paper presents a strategy to integrate the results provided by different EEL approaches to extract and link named entities from unstructured English text. The proposed strategy is composed of three stages to collect, merge and filter such results in order to increase the amount of extractions and reduce possible discrepancies and/or irrelevant extractions without drastically decrease the accuracy of the final result. The proposed strategy differs from existing EEL ensemble-based approaches in the sense that merging and filtering decisions are provided, which are helpful to mitigate the problems previously stated and to easily implement a prototype by leveraging available EEL tools and systems from the literature. Extractions provided by the proposed strategy would benefit applications in areas such as Ontology Learning, Machine Learning and Information Retrieval, to mention a few. Details of the proposed strategy are presented in the following sections.

2 Methodology

The proposed strategy for extracting and linking named entities from unstructured text is based on the integration of the output provided by different EEL

[1] http://wiki.dbpedia.org/

[2] https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/

tools. This strategy consists of three main stages, as shown in Figure 1. These stages are described in the following subsections.

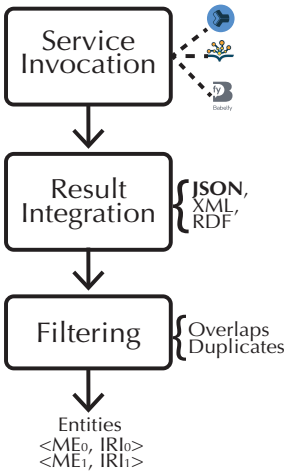

Fig. 1. Methodology stages of the proposed strategy to integrate EEL results

2.1 Service invocation

The main purpose of this stage is the execution of different EEL tools taking into account that these tools can be executed in a local environment (algorithm execution) or remotely (invoking web services via HTTP requests). Thus, the EEL tools have to be already selected for this stage. In recent years, the most common way to make available an EEL tool is through web services, where the invocation of an EEL service is made through an HTTP request, sending one or more parameters like the input text, confidence, output format (e.g. JSON, XML, or RDF), among others to the server. The use of web services instead of standalone applications can be explained by several factors like computational requirements to execute the extraction algorithms, the size of dictionaries or knowledge bases (used to link entities) and to keep private some aspects of their internal processes. Hence, this first stage collects the results from different EEL tools selected beforehand, keeping the original output from each tool.

2.2 Result integration

The result returned by each EEL tool has a predefined number of features, which in most cases are different from one EEL tool to another. To manipulate all the results from the EEL tools it is necessary to merge them. This stage integrates the output from each used EEL tool taking into account the heterogeneity of

features. For example, given the sentence Monterrey city is located in Mexico, the result returned by the DBpedia Spotlight[3] web service for the named entity Monterrey is shown below:


```
{ "@text":"Monterrey city is located in Mexico",
   "@confidence":"0.35",
   "@support":"0",
   "@types":"",
   "@sparql":"",
   "@policy":"whitelist",
   "Resources":[
    {
         "@URI":"http://dbpedia.org/resource/Monterrey",
         "@support":"2558",
         "@types":
             "Schema:Place,DBpedia:Place,DBpedia:PopulatedPlace,
              DBpedia:Settlement,Schema:City,DBpedia:City",
         "@surfaceForm":"Monterrey",
         "@offset":"0",
         "@similarityScore":"0.9934778456173416",
         "@percentageOfSecondRank":"0.005987130977590879"
    },
    {
         "@URI":"http://dbpedia.org/resource/City",
         "@support":"21002","@types":"",
         "@surfaceForm":"city",
         "@offset":"10",
         "@similarityScore":"0.8995032710133117",
         "@percentageOfSecondRank":"0.04685583690072002"
     },
     {
         "@URI":"http://dbpedia.org/resource/Mexico",
         "@support":"82072",
         "@types":
             "Schema:Place,DBpedia:Place,DBpedia:PopulatedPlace,
              Schema:Country,DBpedia:Country",
         "@surfaceForm":"Mexico",
         "@offset":"29",
         "@similarityScore":"0.9974348213727555",
         "@percentageOfSecondRank":"9.413710843046881E-4"}
  ]}
```


The DBpedia Spotlight service returns several features such as *@URI*, which refers to the identifier associated with the mention (the entity); *@support* refers

[3] https://github.com/dbpedia-spotlight/dbpedia-spotlight

to the degree of confidence of the resource assigned to the named entity in the KB; *@surfaceForm* refers to the text of the mention; *@offset* refers to the position of the mention in the input text (number of characters); *@similarityScore* refers to the similarity between the mention and the descriptive label of the resource in the KB; *@percentageOfSecondRank* indicates a degree of support relative to the ambiguity between possible resources allocated to the mention.

Although the names of the returned features vary for each EEL tool, there are common features such as the surface form (string of the entity mention), the position of the mention in the input text (offset), and the identifier (IRI) from the KB. Therefore, this step is intended to match the varied outputs produced by the selected EEL tools in such a way that the three previously mentioned features are obtained and thus, an homogeneous result can be produced.

2.3 Filtering

The final step filters the result to solve issues like duplicates and overlapped entities. For such purpose, three main cases are considered:

1. Duplicate entity mentions. This case occurs when two or more tuples have the same text as entity mention, but different identifier. In this case, a majority voting strategy was implemented to select the most frequent tuple. In the case of a tie, the EEL tools can be manually ranked so that the entity returned by the best ranked tool is selected.
2. Overlap entities. It refers to the case when an entity mention is partially contained or overlapped within another mention. For example, the entity mention Police Department is partially contained in the mention Monterrey Police Department. The proposed strategy is to keep the longest matching named entity with respect to the input text.
3. Duplicate tuples. For this case, duplicate tuples are eliminated, preserving one of the tuples returned by either of the invoked tools.

Finally, the result is stored in JSON format to be used by any application.

3 Results and discussion

This section presents the experiments and results conducted for the proposed strategy. For such purposes, a Java implementation was developed and configured as presented in the following subsections.

3.1 Dataset

The experiments were performed using the testing dataset provided in the Open Knowledge Extraction Challenge (OKE)[4] event. This dataset contains 193 manually labelled sentences with 875 tuples of entities linked to DBpedia. Note that NIL (Not In Lexicon) entities are not considered for these experiments (those entities that are relevant but do not contain an association with a KB resource).

[4] https://github.com/anuzzolese/oke-challenge-2016

3.2 Experiments

Three state of the art EEL tools were employed for the experiments. All of them were invoked via web service: DBpedia Spotlight, Babelfy[5] and TagMe[6]. These tools were selected because they provide configurable and publicly available web services, additionally, they have shown top results as reported in [9].

Each tool was configured using different confidence values in order to obtain a balance between the number of extracted entities and precision. For DBpedia Spotlight the confidence value was set as 0.5, for TagMe as 0.07 and for Babelfy as 0.5. Finally, traditional information retrieval metrics (precision, recall and F-measure (F1)) were used to evaluate the performance of the proposed strategy following an exact matching comparison, i.e., the extracted entity tuples must match exactly to those provided in the testing dataset. The results are shown in Table 1, where the results of each individual EEL tool are compared with respect to the results provided by the proposed integration strategy.

EEL system	Precision	Recall	F1	Extracted entities
DBpedia Spotlight	0.4579	0.4982	0.4772	952
Babelfy	**0.4581**	0.5062	0.4809	967
TagMe	0.4073	0.5954	0.4837	1279
Proposed strategy	0.4152	**0.6354**	**0.5022**	**1339**

Table 1. Results obtained from the entity extraction task comparing each tool versus the proposed strategy.

As shows Table 1, the results of each individual EEL tool are not the best for Precision, Recall and F1 simultaneously. Meanwhile, the proposed strategy obtained the best values for Recall and F1. Moreover, an additional number of entities were extracted than using any single tool.

The proportion of true positives (TP), false positives (FP) and false negatives (FN) obtained by each tool for the testing dataset are shown in Table 2. These values were used to obtain the values depicted in Table 1.

3.3 Discussion

The results presented in the experiments show that the proposed strategy increases the performance with respect to the F1 measure in comparison with the tested EEL tools. However, given that the result provided by the tools is merged into one homogeneous output, the Precision is negatively affected since

[5] http://babelfy.org/
[6] http://tagme.di.unipi.it/

EEL system	TP	FP	FN
DBpedia Spotligh	436	516	**439**
Babelfy	443	524	432
TagMe	521	758	354
Proposed strategy	**556**	**783**	319

Table 2. Comparison between the tuples obtained by each tool and the proposed strategy with respect to the testing dataset.

the False Positive values are accumulated for the final result, which directly affect the Precision of the proposed approach. On the other hand, the output integration produced a greater number of entities linked to DBpedia from unstructured text. In some cases, False Positive named entities were extracted because the testing dataset does not have a link to describe them. For example, given the sentence "*In 1842, a woman graduated with distinction from the National Autonomous University of Mexico*", the proposed approach extracted the mention "*woman*", linked it to the resource *http://dbpedia.org/resource/Woman*, this result is marked as a false positive because this tuple does not appear in the testing dataset.

4 Conclusions

This paper presented a strategy to extract and link named entities from unstructured text. The strategy is based on the integration of three different state of the art EEL tools and the addition of three filtering rules to tackle problems of duplicated and overlapped entity mentions. The results obtained in the experiments demonstrate that the proposed strategy improves the extraction and linking according to the F1 measure and, additionally, the proposed strategy increases the number of entities linked to DBpedia KB in comparison with each EEL tool tested at the cost of a slight impact in the final precision.

References

1. Antoniou, G., van Hermelen, F.: A Semantic Web Primer, Second Edition. The MIT Press (Dec 2008)
2. Auer, S., Bryl, V., Tramp, S. (eds.): Linked Open Data - Creating Knowledge Out of Interlinked Data - Results of the LOD2 Project, Lecture Notes in Computer Science, vol. 8661. Springer (2014)
3. Ceccarelli, D., Lucchese, C., Orlando, S., Perego, R., Trani, S.: Dexter 2.0 - an open source tool for semantically enriching data. In: ISWC-PD. pp. 417–420 (2014)
4. Chabchoub, M., Gagnon, M., Zouaq, A.: Collective disambiguation and semantic annotation for entity linking and typing. In: Third SemWebEval Challenge at ESWC. pp. 33–47. Springer (2016), https://doi.org/10.1007/978-3-319-46565-4_3

5. Haidar-Ahmad, L., Font, L., Zouaq, A., Gagnon, M.: Entity typing and linking using SPARQL patterns and dbpedia. In: Semantic Web Challenges - Third SemWebEval Challenge at ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Revised Selected Papers. pp. 61–75. Springer (2016), https://doi.org/10.1007/978-3-319-46565-4_5
6. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: DBpedia Spotlight: Shedding Light on the Web of Documents. In: International Conference on Semantic Systems. pp. 1–8. I-Semantics '11, ACM, New York, NY, USA (2011), http://doi.acm.org/10.1145/2063518.2063519
7. Piccinno, F., Ferragina, P.: From tagme to wat: A new entity annotator. In: Proceedings of the First International Workshop on Entity Recognition & Disambiguation. pp. 55–62. ERD '14, ACM, New York, NY, USA (2014), http://doi.acm.org/10.1145/2633211.2634350
8. Plu, J., Rizzo, G., Troncy, R.: Enhancing entity linking by combining NER models. In: Semantic Web Challenges - Third SemWebEval Challenge at ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Revised Selected Papers. pp. 17–32. Springer (2016)
9. Röder, M., Usbeck, R., Ngomo, A.N.: Gerbil benchmarking named entity recognition and linking consistently. Semantic Web (2017)
10. Shen, W., Wang, J., Han, J.: Entity linking with a knowledge base: Issues, techniques, and solutions. IEEE Transactions on Knowledge and Data Engineering 27(2), 443–460 (2015)