IRIS&LIMSI at TAC KBP Trilingual Entity Discovery and Linking 2017

Jose G. Moreno, Brigitte Grau

IRIS&LIMSI at TAC KBP Trilingual Entity Discovery and Linking 2017

Jose G. Moreno* and Brigitte Grau**

jose.moreno@irit.fr, brigitte.grau@limsi.fr
*Université de Toulouse UPS-IRIT,118 route de Narbonne F- 31062 Toulouse cedex 9
**LIMSI, CNRS, ENSIIE, F-91405 Orsay, France, Université Paris-Saclay

Abstract. This paper presents the collaborative participation between the IRIS team and the LIMSI laboratory to the Trilingual Entity Discovery and Linking of TAC KBP 2017. The aim of the EDL track is to evaluate systems that automatically detect entities in raw text and manage to linking them. In our first joint participation, we focused on the entity linking task and use a public available software for the entity recognition task. Results show that an extra effort must be performed to improve the entity recognition phase in order to improve the later entity linking step.

1 Introduction

During the 2017 edition of the TAC KBP evaluation campaign, we participated in the Trilingual Entity Discovery and Linking, which goal is to annotate and link mentions of entities in raw texts. Following a common architecture of EDL systems [1], we first identify mentions of entities (Entity Discovery) in raw documents to later link them (Entity Linking) to an entity in a Knowledge Base (KB). In this first participation, we mainly focused on the later and use a standard implementation for the former. Entity linking (EL) consists in accurately identifying entities from a KB mentioned in a previously selected portion of text. Within that context, we experiment a simple but powerful entity linking algorithm for the English language.

Several systems are available in the literature for the EL task, including well-known systems such as Wikify[2], AIDA[3] and Spotlight[4]. They make use of different resources and features to automatically identify entities in text documents. However, their code architecture makes hard to grasp the contribution of each feature. In order to understand the individual contribution of each feature, we opt for a candidate representation in a vector space where each feature is a dimension.

As we are interested in the EL task, we tested our implementation with the TAC KBP 2015 diagnostic task before submitting our run. Results showed that our implementation fairly approximates algorithms of the state-of-the-art. However, the results with 2017 data show a different behavior. We believe that the decrease in our performance is only due to the performance of the named entity recognition system. We publish our system implementation to encourage new research on feature engineering under a supervised setup.

2 Named entity recognition

As mentioned in Section 1, our work was not focused on this step. For that reason, we have applied a standard and public available tool. In particular, we used the ne_chunk method from the nltk[1] package. No special configuration was considered and all parameters were set by default. This method uses multiple resources, the results of a Part-of-Speech tagger and a maximum entropy model trained on the Automatic Content Extraction (ACE) corpus, to predict a mention of an entity and its type. The list of types and its matching to the EDL task are presented in Table 1.

Each entity recognized by the system is stored and used as surface form for the entity linking method described in Section 3.

[1] http://www.nltk.org/

Table 1. ACE to EDL type mapping for the named entity recognition step. The mention is ignored when the ACE types DATE, TIME, MONEY and PERCENT are detected.

ACE type	EDL type
ORGANIZATION	ORG
PERSON	PER
LOCATION	LOC
DATE	-
TIME	-
MONEY	-
PERCENT	-
FACILITY	FAC
GPE	LOC

3 Supervised entity linking

3.1 Problem definition

Given a collection composed by surface forms ($S = s_1, ..., s_m$)[2], their associated entities ($E = e_1, ..., e_m$) and documents where they appear ($D = d_1, ..., d_n$) the supervised entity linking task consists in learning a model to correctly identify the entity e_j for unseen pairs composed by a surface form (s_t) and a document (d_t).

3.2 System implementation

As a preprocessing step, a set of candidate entities must be generated for all mentions in training and test documents. Each candidate (positive or negative) is represented by a vector of features. A model is learned using every positive example (from the training dataset) and a set of randomly selected negative examples from the negative candidates[3]. Once the model is learned, the prediction step consists in classifying each candidate of the test set for a given surface form and selecting the candidate with the highest positive prediction score. Our system works with any classification algorithm able to provide a prediction score as output. In particular, we used a recent and powerful binary classifier, the XGBoost algorithm [5]. If all candidates are predicted as negative classes, then the surface form is considered as a mention of an unknown entity and marked as *NIL*. No extra steps are performed making our system able to fit in few lines of code.

3.3 Surface form similarities for Entity Linking

Fifteen different features were calculated using Lucene[4]. Surface forms in Wikipedia were indexed using Lucene and each substring to disambiguate was used as query. The used features are grouped as title related, anchor text related and ranking features. Additionally, two features related with the popularity of the entity in the KB were added [5]. All of them are listed in Table 2.

[2] The offsets in the corresponding documents.
[3] We use 10 in our experiments.
[4] https://lucene.apache.org/core/
[5] The last two rows in Table 2.

Table 2. Features used for representing each candidate. The source column makes reference to the information used to calculate the feature.

Name	Source	Type
Exact matching	title	binary
Partial matching	title	binary
Jaro-Winkler distance	title	real
Levenshtein distance	title	real
Lucene Levenshtein distance	title	real
N-gram distance	title	real
Exact matching	anchor text	binary
Partial matching	anchor text	binary
Jaro-Winkler distance	anchor text	real
Levenshtein distance	anchor text	real
Lucene Levenshtein distance	anchor text	real
N-gram distance	anchor text	real
TF-IDF score	anchor text	real
Ranking position	anchor text	integer
Frequency	anchor text	integer
Normalized Popularity	entity	real
Normalized Inlink counts	entity	real

4 Experiments and results

4.1 Experiments using the 2015 data

Experiments were performed with a previous published collection from the TAC KBP EDL2015 challenge[1]. We used the data available for the "diagnostic EL track" where entity mentions are provided. Note that in this track, offsets are provided and the named entity recognition subtask is ignored. Training and test collections are composed by 12,175 and 13,587[6] tuples (d_i, s_i and e_i), respectively. We consider the *evaluation* measures proposed by [1]: *strong_typed_link_match* (evaluates linking quality for entities referenced in the KB) and *strong_typed_nil_match* (evaluates the identification of entities that are not part of the KB) and *strong_typed_all_match* (evaluates combined linking quality of the two previous metrics). Table 3 shows results for the two groups of features and their combination. For comparison, results obtained by the best, average, and median participants are included[1]. As expected, the combination of both types of features brings better results than individual groups of features. Moreover, our results using both types outperform the median and average participant results. However, it is still 6 points far from the best participant performance.

Table 3. F-score results for three different feature combinations compared against the best, average and median performances in the TAC KBP 2015 entity linking challenge[1].

Measure	Lucene	Popularity	Lucene+Popularity	Best	Average	Median
strong_typed_link_match	44.9	44.9	69.7	-	-	-
strong_typed_nil_match	54.0	50.2	63.6	74.3	50.8	54.2
strong_typed_all_match	47.8	46.7	67.8	73.9	44.8	45.4

[6] Co-reference tuples are discarded.

Results in Table 3 show that the supervised entity linking task cast well into a classification problem. Indeed, our implementation is a simple classifier but results in a good approximation of the performance of an average participant at EDL 2015.

Recent works in entity linking make extra efforts developing new features and their respective algorithms[6,7,8]. The aim of these first experiments was to show that state-of-the-art performances can be achieved by the use of a simple implementation grounded in the aggregation of multiple features. We make our implementation publicly available at https://github.com/jgmorenof/SupEL to facilitate future research in feature analysis for supervised entity linking.

4.2 Experiments using the 2017 data

Since 2016, the "diagnostic EL" track was removed from the EDL campaign. As a consequence, this year no annotated data was provided. We used the annotated data from the 2015 dataset to learn a model. Predictions over the 2017 dataset were performed with this model. Our results are located in the lower part of the participants ranking. A clear issue in our experiments is the assumption that the ne_chuck method will be able to correctly identify all kind of entity mentions. Indeed, our results in terms of entity discovery are quite low and, as we follow a traditional architecture, the results in terms of entity liking were impacted.

Table 4. Precision, Recall and F-score results of our 2017 participation for the English language. Note that the results of the first three measures for the 2015 data are reported in Table 3. Other measures are included for further comparison.

Measure	Precision	Recall	F-score
strong_typed_link_match	38.6	27.1	31.9
strong_typed_nil_match	11.3	3.4	5.3
strong_typed_all_match	33.7	19.1	24.4
strong_link_match	50.5	35.6	41.7
strong_linked_mention_match	68.1	48.0	56.3
strong_typed_mention_match	47.9	27.2	34.7
strong_nil_match	14.3	4.3	6.6
strong_all_match	44.0	25.0	31.9
entity_match	45.2	50.5	47.7
strong_mention_match	67.0	38.0	48.5
entity_ceaf	27.2	22.4	24.6
mention_ceaf_plus	43.5	24.7	31.5
typed_mention_ceaf	41.8	23.7	30.3
b_cubed_plus	42.6	16.4	23.7
pairwise	89.3	27.2	41.7
muc	78.6	35.3	48.7
typed_mention_ceaf_plus	33.2	18.8	24.0
mention_ceaf	57.5	32.6	41.6
b_cubed	63.3	22.0	32.7

Table 4 shows that the NER system was incapable to retrieve NIL entity mentions. Indeed, only 80 out of 1557 NIL mentions were detected by the NER system. Most of the incorrect predicted NIL mentions are simply incorrect detected mentions. On the contrary, for the *link* measures, only few mentions were not recognized. However, our system wrongly predicted the correct candidate. After double checking our submission, we note that our final run only took into consideration the Lucene features and ignored all the Popularity features. This explain the unexpected under-performance of our system.

5 Conclusion and future work

We presented in this paper our first participation to the TAC KBP EDL track. The results show that a poor performance in the entity recognition task strongly impacts the entity linking results. We intent to apply the lesson learned in this edition for our next participation. In particular, we expect to improve the NER recognizer in order to further improve our final performance.

References

1. Heng Ji, Joel Nothman, Ben Hachey, and Radu Florian. Overview of TAC-KBP2015 Tri-lingual Entity Discovery and Linking. In *Text Analysis Conference (TAC 2015)*, 2015.
2. Rada Mihalcea and Andras Csomai. Wikify!: Linking documents to encyclopedic knowledge. In *Sixteenth ACM Conference, CIKM '07*, pages 233–242, 2007.
3. Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust disambiguation of named entities in text. In *Conference on, EMNLP '11*, pages 782–792, 2011.
4. Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: Shedding light on the web of documents. In *7th International Conference, I-Semantics '11*, pages 1–8, 2011.
5. Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, 2016.
6. Stefan Zwicklbauer, Christin Seifert, and Michael Granitzer. Doser - A knowledge-base-agnostic framework for entity disambiguation using semantic embeddings. In *13th Conference, ESWC 2016*, pages 182–198, 2016.
7. Octavian-Eugen Ganea, Marina Ganea, Aurelien Lucchi, Carsten Eickhoff, and Thomas Hofmann. Probabilistic bag-of-hyperlinks model for entity linking. In *25th International Conference, WWW '16*, pages 927–938, 2016.
8. Jose G. Moreno, Romaric Besançon, Romain Beaumont, Eva D'hondt, Anne-Laure Ligozat, Sophie Rosset, Xavier Tannier, and Brigitte Grau. Combining word and entity embeddings for entity linking. In *The Semantic Web: 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 – June 1, 2017, Proceedings, Part I*, pages 337–352, 2017.