Entity Linking for the Semantic Annotation of Italian Tweets

Pierpaolo Basile*
Università di Bari Aldo Moro

Annalina Caputo**
Università of Bari Aldo Moro

Giovanni Semeraro†
Università di Bari Aldo Moro

Linking entity mentions in Italian tweets to concepts in a knowledge base is a challenging task, due to the short and noisy nature of these short messages and the lack of specific resources for Italian. This paper proposes an adaptation of a general purpose Named Entity Linking algorithm, which exploits the similarity measure computed over a Distributional Semantic Model, in the context of Italian tweets. In order to evaluate the proposed algorithm, we introduce a new dataset of tweets for entity linking that we have developed specifically for the Italian language.

1. Introduction

Twitter, with its average of 500 billion messages being posted every day[1], offers a huge amount of interconnected user generated contents. User mentions, hashtags, and the intricate network that links users together provide a wealth of information about personal interests, social dynamics, and cultural trends that can be mined to benefit Twitter-based tasks, like user profiling and interest discovering, brand analysis and reputation management, tweet and hashtag recommendation, news and trend detection. In order to make such information machine readable and enable the intelligent information access, tools for the extraction and annotation of concepts in tweets are required.

Named Entity Linking (NEL) is the task of semantically annotating entity mentions in a portion of text with links to a knowledge base. This task usually requires as a first step the recognition of portions of text that refer to named entities (*entity recognition*). The linking phase follows, which usually subsumes the entity disambiguation, i.e. selecting the proper concept from a restricted set of candidates (e.g. Mediterraneo <*Movie*> or Mediterraneo <*sea*>). NEL together with Word Sense Disambiguation, i.e. the task of associating each word occurrence with its proper meaning given a sense inventory, is critical to enable automatic systems to make sense of unstructured texts.

Initially developed for reasonably long and clean text (Hoffart et al. 2011, 2012), such as news articles, NEL techniques usually show unsatisfying performance on noisy, short and poorly written text constituted by microblogs such as Twitter (Meij, Weerkamp, and de Rijke 2012; Derczynski et al. 2015). The lack of enough context is one of the main factors that hinders Twitter-based NEL algorithms. The context of an entity

* Department of Computer Science, University of Bari Aldo Moro, Via, E. Orabona, 4 - 70125 Bari (Italy). E-mail: pierpaolo.basile@uniba.it.
** Department of Computer Science, University of Bari Aldo Moro. E-mail: annalina.caputo@uniba.it.
† Department of Computer Science, University of Bari Aldo Moro. E-mail: giovanni.semeraro@uniba.it.
1 http://www.internetlivestats.com/twitter-statistics/

mention usually provides valuable information during the disambiguation step: both the surrounding words and co-occurring entities have been exploited widely in NEL algorithms. Contextual words give a hint about the general topic of a text, e.g. words like *premier*, *festival*, *director* can induce the association between the word *Mediterraneo* and its right entity link to Mediterraneo <*Movie*>. Also, reasoning on related mentions can help during the disambiguation, e.g. the occurrence of other entities like *Europe*, *Africa* or *Loggerhead sea turtle* may promote the link to Mediterraneo <*sea*>. However, Twitter messages may be too short to provide enough contextual evidence, like in the following tweet:

È molto difficile parlare di quello che accade nel mediterraneo ma ci provo sempre.
(It is very difficult to talk about what happens in the Mediterranean, but I always try)

The NEL task becomes even more problematic when the tweet analysis involves languages different from English. For example, out of the ten NER and NEL systems compared on tweets by Derczynski et al. (Derczynski et al. 2015), only four provided support for language different from English. Moreover, many state-of-the-art NEL algorithms usually involve a learning stage (Milne and Witten 2008; Meij, Weerkamp, and de Rijke 2012; Ceccarelli et al. 2013), which in turn requires an annotated corpus. In the context of Italian language, the lack of language-specific resources and annotated tweet datasets hamper the assessment of NEL algorithms for tweets. To overcome these limitations, we propose an unsupervised algorithm for NEL on Twitter and a first manually annotated dataset of Italian tweets. This method is an extension of the original work described in Basile et al. (Basile, Caputo, and Semeraro 2015).

The contributions of this paper to the problem of NEL on Italian tweets are:

- The introduction of an Italian dataset of manually annotated tweets for NEL. To the best of our knowledge, this is the first Italian dataset of such a type. Section 2 reports details concerning the annotation phase and statistics about the dataset.
- The adaptation to Twitter of a NEL algorithm based on a Distributional Semantic Model (DSM-TEL), which needs no specific annotated Italian resources since it is completely unsupervised (Section 3).
- An evaluation of well known NEL algorithms available for the Italian language on this dataset, comparing their performance with our DSM-TEL algorithm in terms of both entity recognition and linking. Section 4 shows and analyses the results of that evaluation.

2. Dataset

One of the main limitations to the development of specific algorithms for tweet-based entity linking in Italian lies on the dearth of datasets for training and assessing the quality of such techniques. Hence, we built a new dataset by following the guidelines proposed in the #Microposts2015 Named Entity Linking (NEEL) challenge[2] (Rizzo et al. 2015). We randomly selected 1,000 tweets from the TWITA dataset (Basile and Nissim 2013), the first corpus of Italian tweets. For each tweet, we first select the named entities (NE). A NE is a string in the tweet representing a proper noun, excluding the preceding article (like "il", "lo", "la", etc.) and any other prefix (e.g. "Dott.", "Prof.") or post-posed

2 http://www.scc.lancs.ac.uk/research/workshops/microposts2015/challenge/

Table 1
The distribution of entities in categories.

Type	*Occurrences*	*Frequency*
Organization	197	0.2606
Character	9	0.0119
Product	96	0.1270
Event	11	0.0146
Person	301	0.3981
Thing	18	0.0238
Location	124	0.1640

modifier. More specifically, an entity is any proper noun that belongs to one of the categories specified in a taxonomy and can either be linked to a DBpedia concept or labelled as NIL, when it has no corresponding concept in DBpedia. The taxonomy is defined by the following categories: Thing[3], Event, Character, Location, Organization, Person and Product.

We annotated concepts by using the canonicalized dataset of Italian DBpedia 2014[4]. For specific Italian concepts that are not linked to an English article, we adopt the localized version of DBpedia. Finally, some concepts have an Italian Wikipedia article but they are not in DBpedia (e.g. *Agorà <TV show>* or *Grazia <magazine>*); in that case we linked the entity by using the Wikipedia URL. Entities represented neither in DBpedia nor Wikipedia are linked to NIL.

The annotation process poses some challenges specific to the Twitter context, where special characters ("#" and "@") identify strings with a specific meaning, i.e. hashtags and user mentions, respectively. For example, all these strings are valid entities: #[Alemanno], and @[CarlottaFerlito]. The '#' and '@' characters are not considered as part of the annotation.

This first version of the dataset was annotated by only one annotator, and comprises 756 entity mentions, with a mean of about 0.75 entities for each tweet. The distribution of entities in categories is reported in Table 1. 63% of tweets links to a DBpedia concept, about 30% of them is annotated as NIL, 6% links to a URL of a Wikipedia page, while only one entity links to an Italian DBpedia concept [5].

The dataset[6] is composed of two files: the data and the annotation file. The data file contains pairs of tweet id and text, each listed on a different line. According to the Twitter license about data, we release only the tweet id and not its content, which can be downloaded by the twitter API. The annotation file consists of a line for each tweet id, which is followed by the start and the end offset[7] of the annotation, the linked concept and the category. All values are separated by the TAB character. For example, for the tweet with id 290460612549545984:

3 Languages, ethnic groups, nationalities, religions, ...
4 This dataset contains triples extracted from Italian Wikipedia articles whose resources have an equivalent English article.
5 `http://it.dbpedia.org/resource/Carlo_Cracco`
6 Available at: `https://github.com/swapUniba/neel-it-twitter`
7 Starting from 0.

@CarlottaFerlito io non ho la forza di alzarmi e prendere il libro! Help me

we have the following annotation:

```
290460612549545984 1 16 http://dbpedia.org/resource/Carlotta_Ferlito Person
```

3. DSM-TEL algorithm

We propose an Entity Linking algorithm specific for Italian tweets that adapts the original method proposed during the NEEL challenge (Basile et al. 2015). Our algorithm consists of two-steps: the initial identification of all possible entity mentions in a tweet followed by the linking of the entities through the disambiguation algorithm. The knowledge base (Wikipedia/DBpedia) is exploited twice in order to 1) extract all the possible surface forms related to entities, and 2) retrieve glosses used during the disambiguation process.

3.1 Entity Recognition

In order to speed up the entity recognition step, we build an index in which for each surface form (entity) the set of all its possible meanings in the knowledge base is reported. Lucene[8] is exploited to build the index where each surface form (entity) is paired with the set of all its possible DBpedia concepts. The surface forms are collected by analysing all the internal links in the Italian Wikipedia dump. Each internal link reports the surface form and the linked Wikipedia page that corresponds to a DBpedia resource. Specifically, for each possible surface form a document composed of two fields is created. The first field stores the surface form, while the second one contains the list of all possible DBpedia concepts that refer to the surface form in the first field. The entity recognition module exploits this index in order to find entities in a text. Given a text fragment, the module performs the following steps:

- Tokenization of the tweet using the Tweet NLP API[9]. We perform some preprocessing operations to manage hashtags and user mentions; for example we split tokens by exploiting upper-case characters: "CarlottaFerlito" $\Longrightarrow$ "Carlotta Ferlito";
- Building all n-grams up to six words;
- Query of the index and retrieval of the top 100 matching surface forms for each candidate entity;
- Scoring each surface form. The score is the linear combination of: a) a string similarity function based on the Levenshtein Distance between the candidate entity and the surface form in the index; b) the Jaccard Index in terms of common words between the candidate entity and the surface form in the index;
- Filtering the candidate entities recognized in the previous steps: entities are removed if the score computed in the previous step is below a given threshold. In this scenario we empirically set the threshold to 0.66;
- Finally, we filter candidate entities according to the percentage of words that: (1) are stop words, (2) are common words[10]; and (3) do not contain at least one upper-

8 http://lucene.apache.org/
9 http://www.ark.cs.cmu.edu/TweetNLP/
10 We exploit the list of 1,000 most frequent Italian words: http://telelinea.free.fr/italien/1000_parole.html

case character. We remove the entity if one of the aforementioned criteria is above the 33%.

The output of the entity recognition module is a list of candidate entities with their set of candidate DBpedia concepts.

3.2 DL-WSD

We exploit the distributional Lesk algorithm proposed by Basile et al. (Basile, Caputo, and Semeraro 2014) for disambiguating named entities. The algorithm replaces the concept of word overlap initially introduced by Lesk (Lesk 1986) with the broader concept of semantic similarity computed in a distributional semantic space. The original Lesk algorithm chooses the proper meaning for a target word on the basis of the word overlaps between the meaning description and the target word context. This method did not acknowledge neither the use of synonyms nor the presence of related words in the context. However, the semantic similarity computed in a distributional space overcomes these limitations by introducing a measure of relatedness between words, which takes into account how similar the words are with respect to their usage in a real corpus.

Let $e_1, e_2, ...e_n$ be a sequence of entity mentions, the algorithm disambiguates each target entity e_i by computing the semantic similarity between the definitions of concepts associated with the target entity and the context of the target. This similarity is computed by representing in a Distributional Semantic Model (DSM) both the definition and the context as the sum of words they are composed of; then this similarity takes into account the co-occurrence evidences previously collected through a corpus of documents. The corpus plays a key role since the richer it is the higher is the probability that each word is fully represented in all its contexts of use. We exploit the word2vec tool[11](Mikolov et al. 2013) in order to build a DSM, by analyzing all the pages in the last Italian Wikipedia Dump[12]. The correct sense for an entity is the one whose gloss maximizes the semantic similarity with the entity context. The algorithm consists of the following steps.

1. **Building the glosses.** We retrieve the set $C_i = \{c_{i1}, c_{i2}, ..., c_{ik}\}$ of concepts associated to the entity e_i. The C_i set is provided by the entity recognition module. For each concept c_{ij}, the algorithm builds the definition representation d_{ij} by summing the vectors of the words that occur in the abstract associated to c_{ij}.
2. **Building the context.** The context T for the entity e_i is represented by all the words that occur in the tweet.
3. **Building the vector representations.** The context T and each definition d_{ij} are represented as vectors in the *WordSpace* built through the DSM. In this space, the vector proximity expresses the semantic similarity between words, traditionally computed as the cosine of the angle between the two word-vectors. The concept of *semantic similarity* can be extended to whole sentences via the vector addition ($+$) operator. A sentence can always be represented as the sum of the word vectors it is composed of. Then, vector addition can be exploited to represent both the definition and the target entity context in order to assess their similarity.
4. **Concept ranking.** The algorithm computes the cosine similarity between the vector representation of each definition d_{ij} and that of the context T. Then, the cosine

11 https://code.google.com/p/word2vec/
12 We use 400 dimensions for vectors analysing only terms that occur at least 25 times.

similarity is linearly combined with a function which takes into account the usage of the concept in Wikipedia. We analyse a function that computes the probability assigned to each concept given an entity (surface form) taking into account the number of times a Wikipedia page is linked from another page.
We define the probability $p(c_{ij}|e_i)$ that takes into account the concept distribution of c_{ij} given the entity e_i. The concept distribution is computed as the number of times the entity e_i is tagged with the concept. Zero probabilities are avoided by introducing an additive (Laplace) smoothing. The probability is computed as follows:

$$p(c_{ij}|e_i) = \frac{t(e_i, c_{ij}) + 1}{\#e_i + |C_i|} \quad (1)$$

where $t(e_i, c_{ij})$ is the number of times the entity e_i is tagged with the concept c_{ij}.

4. Evaluation

The evaluation aims to compare several entity linking tools for Italian tweets by exploiting the proposed dataset. We include in the evaluation our method that is an adaptation of the system that participated in the NEEL challenge for English tweets (Basile et al. 2015).

We select three tools able to perform entity linking for Italian: TAGME, Babelfy, and TextRazor. TAGME (Ferragina and Scaiella 2010) has a particular option that enables a special parser for Twitter messages. This parser has been designed to better handle entities in tweets like URL, user mentions and hash-tag. We enable this option during the evaluation. Some other tools are not developed specifically for Twitter. In particular, Babelfy (Moro, Raganato, and Navigli 2014) is an algorithm for entity linking and disambiguation developed for generic texts that uses BabelNet (Navigli and Ponzetto 2012) as knowledge source. During the evaluation, we set up the Babelfy parameters as follows: annotation type is set to "`NAMED_ENTITIES`", annotation resource is set to "`BN`" and matching type is set to "`PARTIAL_MATCHING`". All the other parameters are left with their default values. The third system is TextRazor[13], a system able to recognize, disambiguate and link entities in ten languages, including Italian. Since TextRazor is a commercial tool (no details about its algorithm are supplied), no customization is available for this system. We provide all the systems with the same input text, on which no preprocessing has been applied.

Systems are compared using the typical metrics of precision (P), recall (R) and F-measure (F). We compute the metrics in two settings: the **exact match** setting requires that both start and end offsets match the gold standard annotation, while in **non exact match** the offsets provided by the systems can differ of two positions with respect to the gold standard.

Each algorithm provides a different output that needs some post-processing operations in order to make it comparable with our annotation standard. Most of the annotations are made with respect to the canonicalized version of DBpedia, while only the 6% of the dataset is annotated using Wikipedia page URLs or the localized version of DBpedia (just one annotation). Babelfy is able to directly provide canonicalized DBpedia URIs. When a BabelNet concept is not mapped to a DBpedia URI, we return

13 `https://www.textrazor.com/`

a NIL instance. TAGME returns Italian Wikipedia page titles that we easily translate into DBpedia URIs. We firstly try to map the page title in the canonicalized DBpedia, otherwise into the Italian one. TextRazor supplies Italian Wikipedia URLs or English Wikipedia URLs that we map to DBpedia URIs. Our algorithm provides Italian DBpedia URIs that we translate into canonicalized URIs when it is possible, otherwise we keep the Italian URIs. To recap: all algorithms are able to provide canonicalized and localized DBpedia URIs, only Babelfy is limited to canonicalized URIs.

4.1 Entity Recognition Evaluation

The entity recognition is a crucial step in the Twitter context, since it is very difficult where no regular language is used like in tweets. In this section we propose a separate evaluation for only the recognition step which takes into account both exact and non exact matches. This evaluation is important for understanding the behavior of each system, since an error in the recognition step compromises the performance during the linking.

Table 2
Results of the entity recognition evaluation with exact and non exact match.

	Exact match			*Non exact match*		
System	P	R	F	P	R	F
Babelfy	.431	.161	.235	.449	.168	.244
TAGME	.363	.458	.405	.391	.492	.436
TextRazor	.605	.310	.410	.605	.310	.410
DSMTEL	.470	.505	**.487**	.495	.532	**.513**

Table 2 reports the results about the entity recognition task with respect to exact and non exact match respectively. DSM-TEL provides the best results followed by TextRazor (exact match) and TAGME (non exact match), while the low performance of Babelfy proves that it is not able to tackle the irregular language used in tweets. In both settings (exact/non exact matching) TextRazor achieves the best precision. Moreover, this is the only system that does not achieve better performance in the non exact match setting, highlighting how this method is able to always detect the proper start and end offset of each entity mention.

For a more accurate analysis, Table 3 and Table 4 show the error rate for each algorithm with respect to the entity categories in both the exact and non exact matches. The most difficult categories are: Character, Event and Thing, which are also the less represented categories (9, 11, 18 instances, respectively). In these classes Babelfy reports an error rate equal to 1, this means that it is not able to recognize any instances belonging to these types. Our algorithm gives the worse performance on the Event category, while it shows a fluctuating behaviour on the other ones. However, we can ascribe its overall better F-measures (Table 2) to the fact that it gives the best performance on the most populous category (Person) combined with the good performances obtained on the other two quite populous categories: Organization and Location.

The easiest category is Location, where TAGME reports an error rate equal to .202 for exact match, and .185 for non exact match.

Table 3
Exact match: error rate on entity recognition.

	Exact match			
	Babelfy	TAGME	TextRazor	DSM-TEL
Organiz.	.838	.523	.604	.538
Character	1	.444	.889	.778
Product	.854	.563	.656	.688
Event	1	.636	.727	.909
Person	.821	.681	.884	.452
Thing	1	.667	.722	.667
Location	.823	.202	.363	.298

Table 4
Non exact match: error rate on entity recognition.

	Non exact match			
	Babelfy	TAGME	TextRazor	DSM-TEL
Organiz.	.827	.497	.604	.513
Character	1	.444	.889	.778
Product	.844	.479	.656	.615
Event	1	.636	.727	.909
Person	.814	.648	.884	.439
Thing	1	.667	.722	.667
Location	.823	.185	.363	.266

4.2 Entity Linking Evaluation

Entity linking performance are reported in Tables 5. It is important to underline that a correct linking requires the proper recognition of the entity involved. TextRazor achieves the best performance in the entity linking task with an F-measure in both exact and non exact matches of 0.280.

Moreover, in order to understand if the recognition and linking tasks pose more challenges for Italian language, we evaluated all the systems on an English dataset. Although the two datasets are not directly comparable (due to the different sizes and the number of entities involved per tweet), we run an experiment over the Making Sense of Microposts (#Microposts2015) Named Entity rEcognition and Linking (NEEL) Challenge dataset (Rizzo et al. 2015) (Table 6). The evaluation results show a different behaviour of the systems for the English language. The recognition task seems more difficult for English than for Italian: all systems except TextRazor obtained lower F-measure figures on #Micropost2015 dataset. Nonetheless, the values of the English linking task are comparable, if not better, than those of the Italian dataset. The only exception is Babelfy that on the English dataset performs poorly also in the linking task. On the English dataset, TextRazor performs the best in both recognition settings, and in the linking with exact match, while the overall best linking performance is obtained by TAGME.

Table 5
Results of the entity linking evaluation with exact and non exact match.

	Exact match			*Non exact match*		
System	P	R	F	P	R	F
Babelfy	.318	.119	.173	.322	.120	.175
TAGME	.226	.284	.252	.235	.296	.262
TextRazor	.413	.212	**.280**	.413	.212	**.280**
DSM-TEL	.245	.263	.254	.254	.272	.263

Table 6
F-Measure results for English #Microposts2015 NEEL dataset.

	Recognition		*Linking*	
System	Exact	No Exact	Exact	No Exact
Babelfy	.134	.137	.102	.104
TAGME	.352	.381	.290	**.311**
TextRazor	**.460**	**.485**	**.294**	.295
DSM-TEL	.442	.467	.284	.299

Also in the linking task we report the error rate for each entity category for exact (Table 7) and non exact matching (Table 8). The hardest entity type to disambiguate is Character, no one algorithm is able to link at least one instance of this type. Location confirmed to be the easiest type also for linking, where TAGME still achieves the best performance.

Table 7
Exact match: error rate on entity linking.

	Exact match			
	Babelfy	TAGME	TextRazor	DSM-TEL
Organiz.	.863	.777	.761	.761
Character	1	1	1	1
Product	.885	.781	.781	.813
Event	1	.727	.727	.909
Person	.884	.794	.927	.748
Thing	1	.944	1	.889
Location	.863	.323	.460	.556

5. Related Work

Named Entity Linking has received much attention within NLP tasks as a way of bringing semantics and structured information into unstructured text. NEL approaches were initially developed for well formatted and formal text, like news articles, where generally the entity mentions are correctly capitalized and are surrounded by enough text that helps during the disambiguation, and where the co-occurrence of many others

Table 8
Non exact match: error rate on entity linking.

	Non exact match			
	Babelfy	TAGME	TextRazor	DSM-TEL
Organiz.	.863	.766	.761	.756
Character	1	1	1	1
Product	.875	.781	.781	.802
Event	1	.727	.727	.909
Person	.884	.781	.927	.741
Thing	1	.944	1	.889
Location	.863	.306	.460	.532

entity mentions has fostered the adoption of graph-based methods. In this scenario, two kinds of approaches have been developed: *local* and *global* disambiguation.

In local approaches, each entity mention is considered individually. Then, the disambiguation algorithm targets one entity at time. The surrounding words of the entity under analysis are usually exploited as a clue to infer the proper entity link. These techniques can be considered as an extension of the classical Lesk algorithm (Lesk 1986) to named entities, and they generally rely on some measures of similarity between the context of the mention and the content description of the candidate entity. The algorithms of Razvan and Bunescu (Bunescu and Pasca 2006) and Mihalcea and Csomai (Mihalcea and Csomai 2007) both exploit some measures of word overlap between context words and the candidate entity Wikipedia page. Similar to their work, the algorithm we propose mainly differs in the use of Distributional Semantics to overcome the non-exact match between words that may occur when similar or related words are used in the entity context/description.

Global disambiguation algorithms attempts to disambiguate all the entity mentions in a text at once. These methods usually exploit some measures of *coherence* between entities, and try to select the group of candidates that maximize the coherence (Cucerzan 2007). Many of these algorithms make use of the Milne and Witten measure (Milne and Witten 2008), which computes the similarity between two entities as a measure of their common ingoing links. Kulkarni et al. (Kulkarni et al. 2009) and Ratinov et al. (Ratinov et al. 2011) both use this measure in hybrid approaches which try to optimize both local and global measures of similarity. Similar to (Kulkarni et al. 2009), TAGME (Ferragina and Scaiella 2010), which has been developed to annotate both long and short texts, is based on a hybrid approach that first disambiguates the mentions, and then it prunes the non pertinent ones. The disambiguation phase is based on the Milne and Witten measure: given a mention, TAGME initially computes the relatedness between each candidate link for that mention and the candidate links of all the other mentions in the same text fragment. Then, it chooses the proper entity link by implementing a voting scheme that computes the vote of any other mention to the annotation. Eventually, some bad annotations are removed through a classifier which exploits as features both the link probability and the coherence measure. A graph-based approach is implemented in Babelfy (Moro, Raganato, and Navigli 2014). Babelfy builds a semantic signature for the nodes in the graph built upon BabelNet: through a random walk with restart each vertex in the graph is associated to a set of related vertexes. This semantic signature is then exploited in the disambiguation phase where a directed graph is built upon the text

fragment to disambiguate; in this graph a connection between two nodes exists if one of them belongs to the semantic signature of the other. The most suitable interpretation for the disambiguation of mentions is obtained by choosing the densest subgraph and measuring the normalized weighted degree of each meaning.

Local and global approaches can be applied to disambiguate short fragments of text, like tweets or microposts, although in such scenarios new challenges may emerge. Derczynski et al. (Derczynski et al. 2015) evaluated ten commercial and research systems for entity recognition and disambiguation with the aim of analysing their performances and pointing out challenges peculiar to Twitter messages. From this study many factors have emerged that can be detrimental to the NEL performance. Shortness and noise are the two main factors, but also multilingual content and references to the user or the social context are elements that influence the linking algorithms. The limit of 140 characters of Twitter messages poses serious challenges to linking algorithms, since the lack of proper context, in terms of both words and other entities, may hamper the inference of a tweet topic. Then, one way to overcome this limitation is by extending a tweet context. Cassidy et al. (Cassidy et al. 2012) experimented with two possible extensions of a tweet content: either via tweets on the same target topic or through other tweets of the same authors. Both approaches improved over the baseline, with the latter performing the best. The idea of expanding the context by tweets of the same authors is also behind the work proposed in (Shen et al. 2013). In this case, the authors build a model of the user's topics of interest, and on this basis they run a propagation algorithm on the graph of the entities of interest. Liu et al. (Liu et al. 2013) approach the lack of context in a different way by assuming the "similar mention with similar entity" principle. They propose a method that tries to disambiguate all mentions at once by using three different measures: 1) a context measure that computes the similarity between the mention context and the candidate entity description, 2) a coherence measure that captures the relatedness between all the candidate entities involved in the text and 3) a measure of the similarity between mentions. The last measure exploits the redundancy of the same mention across different tweets, and it is used to boost the coherence measure between entities. Yerva et al. (Yerva et al. 2013) address the problem of classifying a tweet with respect to an ambiguous company name: here the user profile built upon social networks provides further contextual information for the disambiguation. Another approach to Twitter-based NEL can be that of exploiting specific features. Meij et al. (Meij, Weerkamp, and de Rijke 2012) conducted an analysis of several state-of-the-art entity linking algorithm applied on tweets and compared these baselines with a learning to rank approach where, among the others, they exploited Twitter specific features. Some of these features are at the core of the disambiguation algorithm that performed the best during #Microposts2015 NEEL challenge (Yamada, Takeda, and Takefuji 2015).

6. Conclusion

We tackled the problem of entity linking for Italian tweets. Our contribution is threefold: 1) we built a first Italian tweet dataset for entity linking, 2) we adapted a distributional-based NEL algorithm to the Italian language, and 3) we compared state-of-the-art systems on the built dataset. As for English, the entity linking task for Italian tweets turned out to be quite difficult, as pointed out by the very low performance of all systems employed. As future work we plan to extend the dataset in order to provide more examples for training and testing data.

References

Basile, Pierpaolo, Annalina Caputo, and Giovanni Semeraro. 2014. An Enhanced Lesk Word Sense Disambiguation Algorithm through a Distributional Semantic Model. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1591–1600, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Basile, Pierpaolo, Annalina Caputo, and Giovanni Semeraro. 2015. Entity Linking for Italian Tweets. In Cristina Bosco, Sara Tonelli, and Fabio Massimo Zanzotto, editors, *Proceedings of the Second Italian Conference on Computational Linguistics CLiC-it 2015*, pages 36–40, Trento, Italy, December 3-8. Accademia University Press.

Basile, Pierpaolo, Annalina Caputo, Giovanni Semeraro, and Fedelucio Narducci. 2015. UNIBA: Exploiting a Distributional Semantic Model for Disambiguating and Linking Entities in Tweets. In *Proceedings of the the 5th Workshop on Making Sense of Microposts co-located with the 24th International World Wide Web Conference (WWW 2015)*, volume 1395, pages 62–63. CEUR-WS.

Basile, Valerio and Malvina Nissim. 2013. Sentiment analysis on italian tweets. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 100–107, Atlanta, Georgia, June. Association for Computational Linguistics.

Bunescu, Razvan C. and Marius Pasca. 2006. Using encyclopedic knowledge for named entity disambiguation. In Diana McCarthy and Shuly Wintner, editors, *Proceedings of EACL 2006, 11st Conference of the European Chapter of the Association for Computational Linguistics*, pages 9–16, Trento, Italy, April 3-7. The Association for Computer Linguistics.

Cassidy, Taylor, Heng Ji, Lev-Arie Ratinov, Arkaitz Zubiaga, and Hongzhao Huang. 2012. Analysis and enhancement of wikification for microblogs with context expansion. In *Proceedings of COLING 2012*, pages 441–456, Mumbai, India, December. The COLING 2012 Organizing Committee.

Ceccarelli, Diego, Claudio Lucchese, Salvatore Orlando, Raffaele Perego, and Salvatore Trani. 2013. Learning relatedness measures for entity linking. In Qi He, Arun Iyengar, Wolfgang Nejdl, Jian Pei, and Rajeev Rastogi, editors, *22nd ACM International Conference on Information and Knowledge Management, CIKM'13*, pages 139–148, San Francisco, CA, USA, October 27 - November 1. ACM.

Cucerzan, Silviu. 2007. Large-scale named entity disambiguation based on wikipedia data. In Jason Eisner, editor, *EMNLP-CoNLL 2007, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 708–716, Prague, Czech Republic, June 28-30. ACL.

Derczynski, Leon, Diana Maynard, Giuseppe Rizzo, Marieke van Erp, Genevieve Gorrell, Raphaël Troncy, Johann Petrak, and Kalina Bontcheva. 2015. Analysis of named entity recognition and linking for tweets. *Inf. Process. Manage.*, 51(2):32–49.

Ferragina, Paolo and Ugo Scaiella. 2010. Fast and accurate annotation of short texts with wikipedia pages. *IEEE Software*, 29(1):70–75.

Hoffart, Johannes, Stephan Seufert, Dat Ba Nguyen, Martin Theobald, and Gerhard Weikum. 2012. Kore: Keyphrase overlap relatedness for entity disambiguation. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 545–554, New York, NY, USA. ACM.

Hoffart, Johannes, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 782–792, Edinburgh, United Kingdom. Association for Computational Linguistics.

Kulkarni, Sayali, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. 2009. Collective annotation of wikipedia entities in web text. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 457–466, New York, NY, USA. ACM.

Lesk, Michael. 1986. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation*, SIGDOC '86, pages 24–26, New York, NY, USA. ACM.

Liu, Xiaohua, Yitong Li, Haocheng Wu, Ming Zhou, Furu Wei, and Yi Lu. 2013. Entity linking for tweets. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1304–1311, Sofia, Bulgaria, August. Association for

Computational Linguistics.
Meij, Edgar, Wouter Weerkamp, and Maarten de Rijke. 2012. Adding semantics to microblog posts. In *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, WSDM '12, pages 563–572, New York, NY, USA. ACM.
Mihalcea, Rada and Andras Csomai. 2007. Wikify!: Linking documents to encyclopedic knowledge. In *Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management*, CIKM '07, pages 233–242, New York, NY, USA. ACM.
Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of ICLR Work.*
Milne, David N. and Ian H. Witten. 2008. Learning to link with wikipedia. In James G. Shanahan, Sihem Amer-Yahia, Ioana Manolescu, Yi Zhang, David A. Evans, Aleksander Kolcz, Key-Sun Choi, and Abdur Chowdhury, editors, *Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008*, pages 509–518, Napa Valley, California, USA, October 26-30. ACM.
Moro, Andrea, Alessandro Raganato, and Roberto Navigli. 2014. Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, 2:231–244.
Navigli, Roberto and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.
Ratinov, Lev, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1375–1384, Portland, Oregon. Association for Computational Linguistics.
Rizzo, Giuseppe, Amparo Elizabeth Cano Basave, Bianca Pereira, Andrea Varga, Matthew Rowe, Milan Stankovic, and Aba-Sah Dadzie. 2015. Making Sense of Microposts (#Microposts2015) Named Entity rEcognition and Linking (NEEL) Challenge. In *Proceedings of the the 5th Workshop on Making Sense of Microposts co-located with the 24th International World Wide Web Conference (WWW 2015)*, volume 1395, pages 44–53. CEUR-WS.
Shen, Wei, Jianyong Wang, Ping Luo, and Min Wang. 2013. Linking named entities in tweets with knowledge base via user interest modeling. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 68–76, New York, NY, USA. ACM.
Yamada, Ikuya, Hideaki Takeda, and Yoshiyasu Takefuji. 2015. An end-to-end entity linking approach for tweets. In Matthew Rowe, Milan Stankovic, and Aba-Sah Dadzie, editors, *Proceedings of the the 5th Workshop on Making Sense of Microposts co-located with the 24th International World Wide Web Conference (WWW 2015)*, volume 1395 of *CEUR Workshop Proceedings*, pages 55–56, Florence, Italy, May 18th. CEUR-WS.org.
Yerva, Surender Reddy, Michele Catasta, Gianluca Demartini, and Karl Aberer. 2013. Entity disambiguation in tweets leveraging user social profiles. In *IEEE 14th International Conference on Information Reuse & Integration, IRI 2013*, pages 120–128, San Francisco, CA, USA, August 14-16. IEEE.