MicroNeel: Combining NLP Tools to Perform Named Entity Detection and Linking on Microposts

Francesco Corcoglioniti, Alessio Palmero Aprosio, Yaroslav Nechaev, Claudio Giuliano
Fondazione Bruno Kessler
Trento, Italy
{corcoglio,aprosio,nechaev,giuliano}@fbk.eu

Abstract

English. In this paper we present the MicroNeel system for Named Entity Recognition and Entity Linking on Italian microposts, which participated in the NEEL-IT task at EVALITA 2016. MicroNeel combines The Wiki Machine and Tint, two standard NLP tools, with comprehensive tweet preprocessing, the Twitter-DBpedia alignments from the Social Media Toolkit resource, and rule-based or supervised merging of produced annotations.

Italiano. *In questo articolo presentiamo il sistema MicroNeel per il riconoscimento e la disambiguazione di entità in micropost in lingua Italiana, con cui abbiamo partecipato al task NEEL-IT di EVALITA 2016. MicroNeel combina The Wiki Machine e Tint, due sistemi NLP standard, con un preprocessing esteso dei tweet, con gli allineamenti tra Twitter e DBpedia della risorsa Social Media Toolkit, e con un sistema di fusione delle annotazioni prodotte basato su regole o supervisionato.*

1 Introduction

Microposts, i.e., brief user-generated texts like tweets, checkins, status messages, etc., are a form of content highly popular on social media and an increasingly relevant source for information extraction. The application of Natural Language Processing (NLP) techniques to microposts presents unique challenges due to their informal nature, noisiness, lack of sufficient textual context (e.g., for disambiguation), and use of specific abbreviations and conventions like #hashtags, @user mentions, retweet markers and so on. As a consequence, standard NLP tools designed and trained on more 'traditional' formal domains, like news article, perform poorly when applied to microposts and are outperformed by NLP solutions specifically-developed for this kind of content (see, e.g., Bontcheva et al. (2013)).

Recognizing these challenges and following similar initiatives for the English language, the NEEL-IT[1] task (Basile et al., 2016a) at EVALITA 2016[2] (Basile et al., 2016b) aims at promoting the research on NLP for the analysis of microposts in the Italian language. The task is a combination of Named Entity Recognition (NER), Entity Linking (EL), and Coreference Resolution for Twitter tweets, which are short microposts of maximum 140 characters that may include hashtags, user mentions, and URLs linking to external Web resources. Participating systems have to recognize mentions of named entities, assign them a NER category (e.g., person), and disambiguate them against a fragment of DBpedia containing the entities common to the Italian and English DBpedia chapters; unlinked (i.e., NIL) mentions have finally to be clustered in coreference sets.

In this paper we present our MicroNeel system that participated in the NEEL-IT task. With MicroNeel, we investigate the use on microposts of two standard NER and EL tools – The Wiki Machine (Palmero Aprosio and Giuliano, 2016) and Tint (Palmero Aprosio and Moretti, 2016) – that were originally developed for more formal texts. To achieve adequate performances, we complement them with: (i) a preprocessing step where tweets are enriched with semantically related text, and rewritten to make them less noisy; (ii) a set of alignments from Twitter user mentions to DBpedia entities, provided by the Social Media Toolkit (SMT) resource (Nechaev et al., 2016); and (iii) rule-based and supervised mechanisms for merging the annotations produced by NER, EL, and SMT, resolving possible conflicts.

[1] http://neel-it.github.io/

[2] http://www.evalita.it/2016

In the remainder of the paper, Section 2 introduces the main tools and resources we used. Section 3 describes MicroNeel, whose results at NEEL-IT and their discussions are reported in Sections 4 and 5. Section 6 presents the system open-source release, while Section 7 concludes.

2 Tools and Resources

MicroNeel makes use of a certain number of resources and tools. In this section, we briefly present the main ones used in the annotation process. The description of the rest of them (mainly used for preprocessing) can be found in Section 3.

2.1 The Wiki Machine

The Wiki Machine[3] (Palmero Aprosio and Giuliano, 2016) is an open source Entity Linking tool that automatically annotates a text with respect to Wikipedia pages. The output is provided through two main steps: entity identification, and disambiguation. The Wiki Machine is trained using data extracted from Wikipedia and is enriched with Airpedia (Palmero Aprosio et al., 2013), a dataset built on top of DBpedia (Lehmann et al., 2015) that increase its coverage over Wikipedia pages.

2.2 Tint

Tint[4] (Palmero Aprosio and Moretti, 2016) is an easy-to-use set of fast, accurate and extensible Natural Language Processing modules for Italian. It is based on Stanford CoreNLP[5] and is distributed open source. Among other modules, the Tint pipeline includes tokenization, sentence splitting, part-of-speech tagging and NER.

2.3 Social Media Toolkit

Social Media Toolkit[6] (Nechaev et al., 2016), or SMT, is an API that is able to align any given knowledge base entry to a corresponding social media profile (if it exists). The reverse alignment is achieved by using a large database (∼1 million entries) of precomputed alignments between DBpedia and Twitter. SMT is also able to classify any Twitter profile as a person, organization, or other.

3 Description of the System

MicroNeel accepts a micropost text as input, which may include hashtags, mentions of Twitter users, and URLs. Alternatively, a tweet ID can be supplied in input (as done in NEEL-IT), and the system retrieves the corresponding text and metadata (e.g., author information, date and time, language) from Twitter API, if the tweet has not been deleted by the user or by Twitter itself.

Processing in MicroNeel is structured as a pipeline of three main steps, outlined in Figure 1: *preprocessing*, *annotation*, and *merging*. Their execution on an example tweet is shown in Figure 2.

3.1 Preprocessing

During the first step, the *original text* of the micropost is rewritten, keeping track of the mappings between original and rewritten offsets. The *rewritten text* is obtained by applying the following transformations:

- Hashtags in the text are replaced with their tokenizations. Given an hashtag, a bunch of 100 tweets using it is retrieved from Twitter. Then, when some camel-case versions of that hashtag are found, tokenization is done based on the sequence of uppercase letters used.
- User mentions are also replaced with their tokenizations (based on camel-case) or the corresponding display names, if available.
- Slangs, abbreviations, and some common typos (e.g., e' instead of è) in the text are replaced based on a custom dictionary (for Italian, we extracted it from the Wikipedia page `Gergo_di_Internet`[7]).
- URLs, emoticons, and other unprocessable sequences of characters in the text are discarded.
- True-casing is performed to recover the proper word case where this information is lost (e.g., all upper case or lower case text). This task employs a dictionary, which for Italian is derived from Morph-It! (Zanchetta and Baroni, 2005).

To help disambiguation, the rewritten text is then augmented with a textual *context* obtained by aggregating the following contents, if available:

- Hashtag descriptions from *tagdef*[8], a collaborative online service;

[3] http://thewikimachine.fbk.eu/

[4] http://tint.fbk.eu/

[5] http://stanfordnlp.github.io/CoreNLP/

[6] http://alignments.futuro.media/

[7] https://it.wikipedia.org/wiki/Gergo_di_Internet

[8] https://www.tagdef.com/

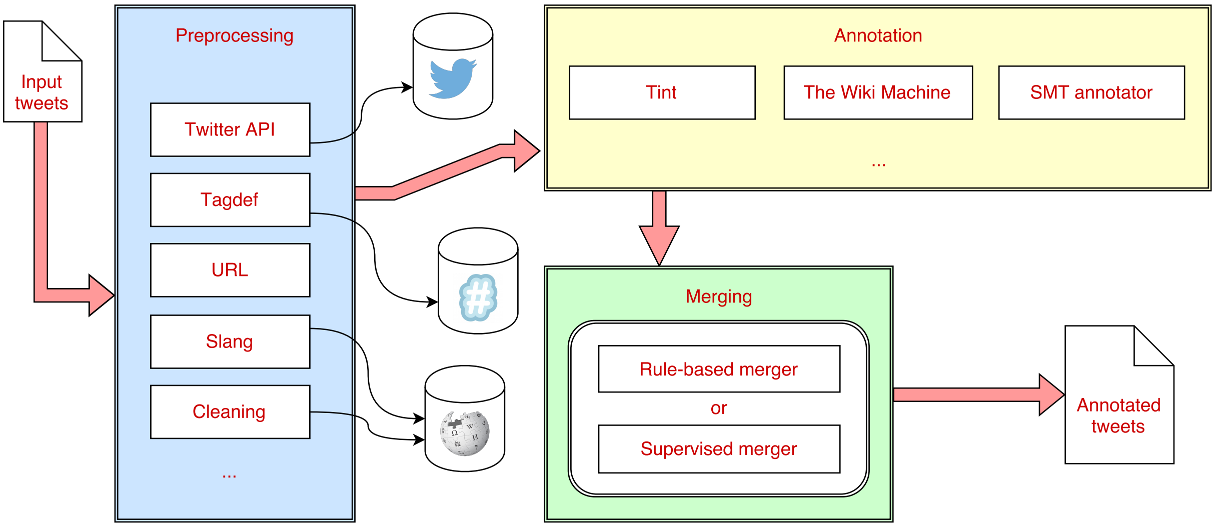


Figure 1: The overview of the system.

- Twitter user descriptions for author and user mentions in the original text;
- Titles of web pages linked by URLs in the original text.

In the example shown in Figure 2, from the original tweet

> **[Original text]**
> (author: @OscardiMontigny)
> #LinkedIn: 200 milioni di iscritti, 4 milioni in Italia http://t.co/jK8MRiaS via @vincos

we collect

- metadata information for the author (Twitter user @OscardiMontigny);
- description of the hashtag #LinkedIn;
- title of the URL http://t.co/jK8MRiaS;
- metadata information for the Twitter user @vincos, mentioned in the tweet.

The resulting (cleaned) tweet is

> **[Rewritten text]**
> LinkedIn: 200 milioni di iscritti, 4 milioni in Italia via Vincenzo Cosenza

with context

> **[Context]**
> Speaker; Blogger; Mega-Trends, Marketing and Innovation Divulgator. #linkedin is about all things from Linkedin. LinkedIn: 200 milioni di iscritti, 4 milioni in Italia — Vincos Blog. Strategist at @BlogMeter My books: Social Media ROI — La società dei dati.

3.2 Annotation

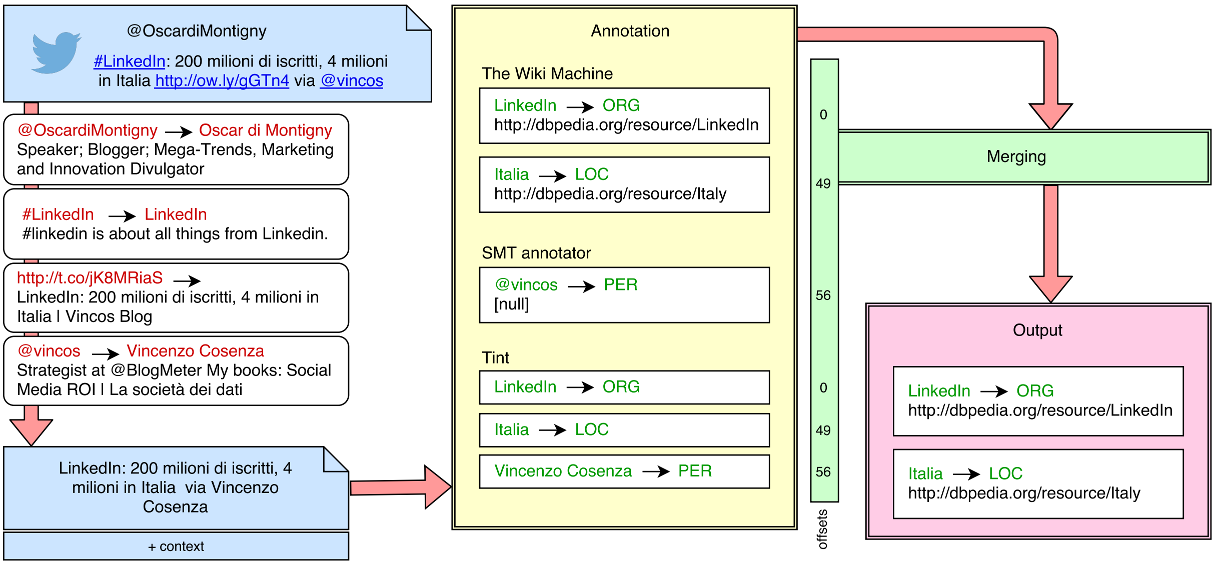
In the second step, annotation is performed by three independent annotator tools run in parallel:

- The rewritten text is parsed with the NER module of Tint (see Section 2.2). This processing annotates named entities of type person, organization, and location.

- The rewritten text, concatenated with the context, is annotated by The Wiki Machine (see Section 2.1) with a list of entities from the full Italian DBpedia. The obtained EL annotations are enriched with the DBpedia class (extended with Airpedia), and mapped to the considered NER categories (person, organization, location, product, event).

- The user mentions in the tweet are assigned a type and are linked to the corresponding DBpedia entities using SMT (see Section 2.3); as for the previous case, SMT types and DBpedia classes are mapped to NER categories. A problem here is that many user mentions classified as persons or organizations by SMT are non-annotable according to NEEL-IT guidelines.[9] Therefore, we implemented two strategies for deciding whether to annotate a user mention:

[9] Basically, a user mention can be annotated in NEEL-IT if its NER category can be determined by just looking at the username and its surrounding textual context in the tweet. Usernames resembling a person or an organization name are thus annotated, while less informative usernames are not marked as their nature cannot be determined without looking at their Twitter profiles or at the tweets they made, which is done instead by SMT.



@OscardiMontigny
#LinkedIn: 200 milioni di iscritti, 4 milioni in Italia http://ow.ly/gGTn4 via @vincos
@OscardiMontigny → Oscar di Montigny
Speaker; Blogger; Mega-Trends, Marketing and Innovation Divulgator
#LinkedIn → LinkedIn
#linkedin is about all things from Linkedin.
http://t.co/jK8MRiaS →
LinkedIn: 200 milioni di iscritti, 4 milioni in Italia I Vincos Blog
@vincos → Vincenzo Cosenza
Strategist at @BlogMeter My books: Social Media ROI I La società dei dati
LinkedIn: 200 milioni di iscritti, 4 milioni in Italia via Vincenzo Cosenza
+ context
Annotation
The Wiki Machine
LinkedIn → ORG
http://dbpedia.org/resource/LinkedIn
Italia → LOC
http://dbpedia.org/resource/Italy
SMT annotator
@vincos → PER
[null]
Tint
LinkedIn → ORG
Italia → LOC
Vincenzo Cosenza → PER
0
49
56
0
49
56
offsets
Merging
Output
LinkedIn → ORG
http://dbpedia.org/resource/LinkedIn
Italia → LOC
http://dbpedia.org/resource/Italy

Figure 2: An example of annotation.

the *rule-based SMT annotator* always annotates if the SMT type is person or organization, whereas the *supervised SMT annotator* decides using an SVM classifier trained on the development set of NEEL-IT.

The middle box in Figure 2 shows the entities extracted by each tool: The Wiki Machine recognizes "LinkedIn" as organization and "Italia" as location; SMT identifies "@vincos" as a person; and Tint classifies "LinkedIn" as organization and "Italia" and "Vincenzo Cosenza" as persons.

3.3 Merging

The last part of the pipeline consists in deciding which annotations have to be kept and which ones should be discarded. In addition, the system has to choose how to deal with conflicts (for example inconsistency between the class produced by Tint and the one extracted by The Wiki Machine).

Specifically, the task consists in building a *merger* that chooses at most one NER class (and possibly a compatible DBpedia link) for each offset of the text for which at least one annotator recognized an entity. For instance, in the example of Figure 2, the merger should ignore the annotation of @vincos, as it is non considered a named entity.

As baseline, we first developed a *rule-based merger* that does not discard any annotation and solves conflicts by majority vote or, in the event of a tie, by giving different priorities to the annotations produced by each annotator.[10]

We then trained a *supervised merger* consisting of a multi-class SVM whose output is either one of the NER categories or a special NONE category, for which case we discard all the annotations for the offset. The classifier is trained on the development tweets provided by the task organizers, using libSVM (Chang and Lin, 2011) with a polynomial kernel and controlling precision/recall via the penalty parameter C for the NONE class. Given an offset and the associated entity annotations we use the following features:

- whether the entity is linked to DBpedia;
- whether the tool x annotated this entity;
- whether the tool x annotated the entity with category y (x can be Tint, SMT, or The Wiki-Machine; y can be one of the possible categories, such as person, location, and so on);
- the case of the annotated text (uppercase initials, all uppercase, all lowercase, etc.);
- whether the annotation is contained in a Twitter username and/or in a hashtag;
- whether the annotated text is an Italian common word and/or a known proper name; common words were taken from Morph-It! (see Section 3.1), while proper nouns were extracted from Wikipedia biographies;
- whether the annotated text contains more than one word;
- frequencies of NER categories in the training dataset of tweets.

The result of the merging step is a set of NER and EL annotations as required by the NEEL-IT

[10] Tint first, followed by The Wiki Machine and SMT.

task. EL annotations whose DBpedia entities are not part of the English DBpedia were discarded when participating in the task, as for NEEL-IT rules. They were however exploited for placing the involved entities in the same coreference set. The remaining (cross-micropost) coreference annotations for unlinked (NIL) entities were derived with a simple baseline that always put entities in different coreference sets.[11]

4 Results

Table 1 reports on the performances obtained by MicroNeel at the NEEL-IT task of EVALITA 2016, measured using three sets of Precision (P), Recall (R), and F1 metrics (Basile et al., 2016a):

- *mention CEAF* tests coreference resolution;
- *strong typed mention match* tests NER (i.e., spans and categories of annotated entities);
- *strong link match* assesses EL (i.e., spans and DBpedia URIs of annotated entities).

Starting from their F1 scores, an overall F1 score was computed as a weighted sum (0.4 for mention CEAF and 0.3 for each other metric).

MicroNeel was trained on the development set of 1000 annotated tweets distributed as part of the task, and tested on 300 tweets. We submitted three runs (upper part of Table 1) that differ on the techniques used – rule-based vs supervised – for the SMT annotator and the merger:

- *base* uses the rule-based variants of the SMT annotator and the merger;
- *merger* uses the rule-based SMT annotator and the supervised merger;
- *all* uses the supervised variants of the SMT annotator and the merger.

In addition to the official NEEL-IT scores, the lower part of Table 1 reports the result of an ablation test that starts from the *base* configuration and investigates the contributions of different components of MicroNeel: The Wiki Machine (EL), Tint (NER), SMT, the tweet rewriting, and the addition of textual context during preprocessing.

5 Discussion

Contrarily to our expectations, the *base* run using the simpler *rule-based SMT* and *rule-based merger* performed better than the other runs employing supervised techniques. Table 1 shows that the contribution of the *supervised SMT* annotator was null on the test set. The *supervised merger*, on the other hand, is only capable of changing the precision/recall balance (which was already good for the *base* run) by keeping only the best annotations. We tuned it for maximum F1 via cross-validation on the development set of NEEL-IT, but the outcome on the test set was a decrease of recall not compensated by a sufficient increase of precision, leading to an overall decrease of F1.

The ablation test in the lower part of Table 1 shows that the largest drop in performances results from removing The Wiki Machine, which is thus the annotator most contributing to overall performances, whereas SMT is the annotator giving the smallest contribution (which still amounts to a valuable +0.0193 F1). The rewriting of tweet texts accounts for +0.0531 F1, whereas the addition of textual context had essentially no impact on the test set, contrarily to our expectations.

An error analysis on the produced annotations showed that many EL annotations were not produced due to wrong word capitalization (e.g., lower case words not recognized as named entities), although the true-casing performed as part of preprocessing mitigated the problem. An alternative and possibly more robust solution may be to retrain the EL tool not considering letter case.

6 The tool

The MicroNeel extraction pipeline is available as open source (GPL) from the project website.[12] It is written in Java and additional components for preprocessing, annotation, and merging can be easily implemented by implementing an `Annotator` interface. The configuration, including the list of components to be used and their parameters, can be set through a specific JSON configuration file. Extensive documentation will be available soon on the project wiki.

7 Conclusion and Future Work

In this paper we presented MicroNeel, a system for Named Entity Recognition and Entity Linking on Italian microposts. Our approach consists of three main steps, described in Section 3: preprocessing, annotation, and merging. By getting the second best result in the NEEL-IT task at EVALITA

[11] It turned out after the evaluation that the alternative baseline that corefers entities with the same (normalized) surface form performed better on NEEL-IT test data.

[12] https://github.com/fbk/microneel

Table 1: MicroNeel performances on NEEL-IT test set for different configurations.

Configuration	Mention CEAF			Strong typed mention match			Strong link match			Overall
	P	R	F1	P	R	F1	P	R	F1	F1
base run	0.514	0.547	0.530	0.457	0.487	0.472	0.567	0.412	0.477	0.4967
merger run	0.576	0.455	0.509	0.523	0.415	0.463	0.664	0.332	0.442	0.4751
all run	0.574	0.453	0.506	0.521	0.412	0.460	0.670	0.332	0.444	0.4736
base - NER	0.587	0.341	0.431	0.524	0.305	0.386	0.531	0.420	0.469	0.4289
base - SMT	0.504	0.525	0.514	0.448	0.468	0.458	0.564	0.372	0.448	0.4774
base - EL	0.487	0.430	0.457	0.494	0.437	0.464	0.579	0.049	0.090	0.3490
base - rewriting	0.554	0.399	0.464	0.492	0.356	0.413	0.606	0.354	0.447	0.4436
base - context	0.513	0.547	0.530	0.453	0.485	0.468	0.566	0.416	0.480	0.4964

2016, we demonstrated that our approach is effective even if it builds on standard components.

Although the task consists in annotating tweets in Italian, MicroNeel is largely agnostic with respect to the language, the only dependencies being the dictionaries used for preprocessing, as both The Wiki Machine and Tint NER support different languages while SMT is language-independent. Therefore, MicroNeel can be easily adapted to other languages without big effort.

MicroNeel is a combination of existing tools, some of which already perform at state-of-the-art level when applied on tweets (for instance, our system got the best performance in the linking task thanks to The Wiki Machine). In the future, we plan to adapt MicroNeel to English and other languages, and to integrate some other modules both in the preprocessing and annotation steps, such the NER system expressly developed for tweets described by Minard et al. (2016).

Acknowledgments

The research leading to this paper was partially supported by the European Union's Horizon 2020 Programme via the SIMPATICO Project (H2020-EURO-6-2015, n. 692819).

References

Pierpaolo Basile, Annalina Caputo, Anna Lisa Gentile, and Giuseppe Rizzo. 2016a. Overview of the EVALITA 2016 Named Entity rEcognition and Linking in Italian tweets (NEEL-IT) task. In *Proceedings of the 5th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2016)*. aAcademia University Press.

Pierpaolo Basile, Franco Cutugno, Malvina Nissim, Viviana Patti, and Rachele Sprugnoli. 2016b. EVALITA 2016: Overview of the 5th evaluation campaign of natural language processing and speech tools for Italian. aAcademia University Press.

Kalina Bontcheva, Leon Derczynski, Adam Funk, Mark A. Greenwood, Diana Maynard, and Niraj Aswani. 2013. TwitIE: An open-source information extraction pipeline for microblog text. In *Recent Advances in Natural Language Processing, RANLP*, pages 83–90. RANLP 2013 Organising Committee / ACL.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. 2015. DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 6(2):167–195.

Anne-Lyse Minard, Mohammed R.H. Qwaider, and Bernardo Magnini. 2016. FBK-NLP at NEEL-IT: Active Learning for Domain Adaptation. In *Proceedings of the 5th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian (EVALITA 2016)*. aAcademia University Press.

Yaroslav Nechaev, Francesco Corcoglioniti, and Claudio Giuliano. 2016. Linking knowledge bases to social media profiles. `http://alignments.futuro.media/`.

Alessio Palmero Aprosio and Claudio Giuliano. 2016. The Wiki Machine: an open source software for entity linking and enrichment. *ArXiv e-prints*.

Alessio Palmero Aprosio and Giovanni Moretti. 2016. Italy goes to Stanford: a collection of CoreNLP modules for Italian. *ArXiv e-prints*.

Alessio Palmero Aprosio, Claudio Giuliano, and Alberto Lavelli. 2013. Automatic expansion of DBpedia exploiting Wikipedia cross-language information. In *Proceedings of the 10th Extended Semantic Web Conference*, pages 397–411. Springer.

Eros Zanchetta and Marco Baroni. 2005. Morph-it! a free corpus-based morphological resource for the Italian language. *Corpus Linguistics 2005*, 1(1).