Tint, the Swiss-Army Tool for Natural Language Processing in Italian

Alessio Palmero Aprosio[0000−0002−1484−0882]

Fondazione Bruno Kessler, Trento, Italy
aprosio@fbk.eu

Abstract. In this we paper present the last version of Tint, an open-source, fast and extendable Natural Language Processing suite for Italian based on Stanford CoreNLP. The new release includes a set of text processing components for fine-grained linguistic analysis, from tokenization to relation extraction, including part-of-speech tagging, morphological analysis, lemmatization, multi-word expression recognition, dependency parsing, named-entity recognition, keyword extraction, and much more. Tint is written in Java freely distributed under the GPL license. Although some modules do not perform at a state-of-the-art level, Tint reaches very good accuracy in all modules, and can be easily used out-of-the-box.

Keywords: Natural Language Processing · Artificial Intelligence · Text Analysis · Readability

1 Introduction

In this paper, we present Tint, a suite of ready-to-use modules for Italian NLP. Tint is free to use, open source, and can be downloaded and used out-of-the-box (see Section 5). Compared to the previous versions [26, 27], the suite has been enriched with several modules for fine-grained linguistic analysis that were not available for Italian before (for example, constituency parsing, relation extraction, temporal expression extraxtion) Finally, some other modules have been trained with new data (named-entity recognition and dependency parsing).

2 Related work

Most of the linguistic pipelines freely available for download (such as Stanford CoreNLP[1] and OpenNLP[2]) are language independent and, even if they are not

[1] http://stanfordnlp.github.io/CoreNLP/
[2] https://opennlp.apache.org/

available in Italian out-of-the-box, they could be trained in every existing language. A notable examples in this direction are UDpipe [32], SpaCy [18], and Stanza [29], trainable pipelines which perform most of the common NLP tasks and are available in almost all the languages included in the Universal Dependencies [20], Freeling [25], a C++ library providing language analysis functionalities for a variety of languages. There are also some other pipelines specifically written for Italian, such as TextPro [14], and T2K [13], but none of them are released as open source (and TextPro is the only one that can be downloaded and used for free for research purposes).

3 Modules

The Tint software is built on top of Stanford CoreNLP, a framework that helps users to derive linguistic annotations for text. Differently from some similar tools, such as UIMA [15] and GATE [9], CoreNLP requires only basic object-oriented programming skills to extend it. The centerpiece of CoreNLP is the pipeline, that takes in raw text, runs a series of NLP annotators on the text, and produces a final set of annotations. CoreNLP supports six languages (Arabic, Chinese, English, French, German, and Spanish). In Tint, we use the CoreNLP paradigm and bring it to Italian.

Among the modules, some of them have been implemented from scratch and do not rely on the components available in Stanford CoreNLP (for example: tokenization, morphological analysis, and so on). Other tasks, such as POS tagging and dependency parsing, are performed using the existing modules in CoreNLP, trained on Italian dataset available to the community. Finally, additional modules include wrappers for existing tools written in Java or available through a web API, such as keyword extraction and entity linking.

We mark with an asterisk (*) the modules that have never been described in a previous work, and with a dagger (†) the ones that were retrained or significantly renovated w.r.t. the past articles.

3.1 Tokenizer and sentence splitter

This module provides text segmentation in tokens and sentences. At first, the text is grossly tokenized. Then, in a second step, tokens that need to be put together are merged using two customizable lists of Italian non-breaking abbreviations (such as “dott.” or “S.p.A.”) and regular expressions (for e-mail addresses, web URIs, numbers, emoticons). This second phase uses [11] to speedup the process.

3.2 Truecaser (*)

The truecase module recognizes the “true” case of tokens (how it would be capitalized in well-edited text) when this information is lost, e.g., all upper case text.

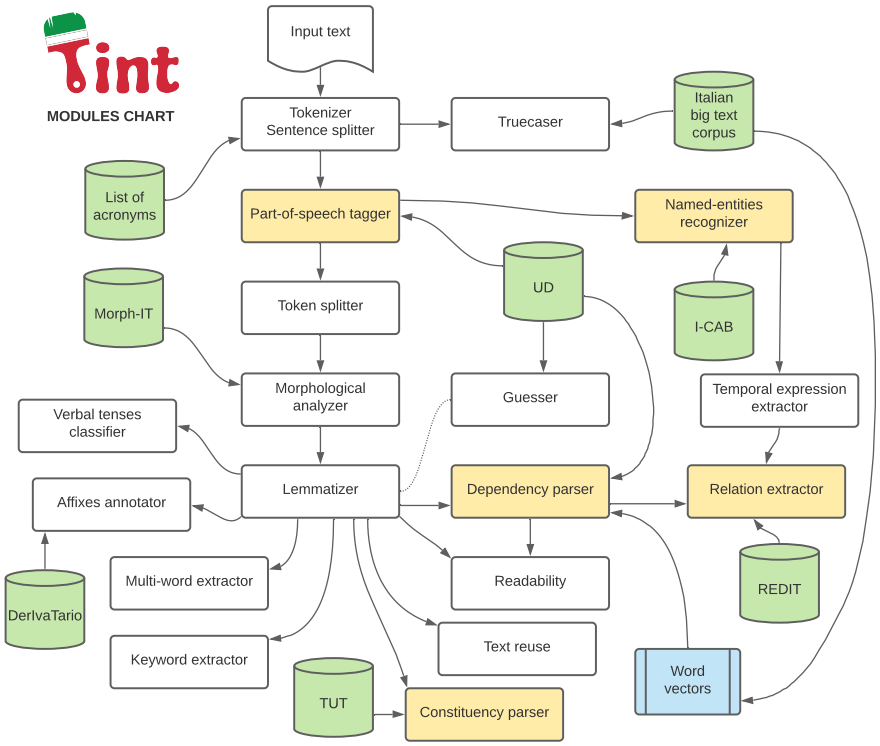

Fig. 1. Tint modules chart. Squared boxes represent modules, and the yellow background means the use of machine learning algorithms. Green blocks indicate linguistic resources. Word vectors are depicted in blue.

It is included in the CoreNLP original package,[3] and relies on a discriminative model using the CRF sequence tagger. The model shipped with Tint has been trained on an Italian corpus (1.3 billion words) that includes texts from different domains: legal, narrative, news, and so on [37].

3.3 Part-of-speech tagger

The part-of-speech annotation is provided through the Maximum Entropy implementation [39] included in Stanford CoreNLP. The model is trained on the Universal Dependencies (UD) dataset for Italian [6].

3.4 Token splitter (*)

The tokenizer (see Section 3.1) is able to split a text into words, but the basic unit for a good morphological annotation is the *syntactic word*. This means

[3] https://stanfordnlp.github.io/CoreNLP/truecase.html

that we systematically want to split off clitics, as in "dammelo" (verb "dà", plus pronouns "me", and "lo"), and undo contractions, as in "alla", that is "a" (preposition) plus "la" (determiner). We refer to such cases as multiword tokens because a single orthographic token corresponds to multiple (syntactic) words.

The CoreNLP pipeline performs such task immediately after segmentation. However, in Italian the tokenization alone is not enough to understand whether a particular token needs to be split. For instance, depending on the context, "delle" can be both a partitive article and a contraction of a preposition and a determiner. Similarly, "porci" can be a noun or a verb plus clitic. We then write a new module for the purpose, using the information provided by the POS module to discriminate the ambiguous cases.

To ensure compatibility with the previous versions of Tint, all the modules provided in Tint that operate after part-of-speech are configured to work either with and without the splitter. Modules that need the training of a model (such as dependency/constituency parsing and named-entity recognition) are trained in both setups: the two models are included in the Tint distribution (one can activate the right one in the configuration file).

3.5 Morphological analyzer

The morphological analyzer module provides the full list of morphological features for each annotated token. The current version of the module has been trained using the Morph-it lexicon [41], but it is possible to extend or retrain it with other Italian datasets. To extend the coverage of the results, especially for the complex forms, such as "porta-ce-ne" or "bi-direzionale", the module decomposes the token into prefix-root-infix-suffix and tries to recognise the root form.

3.6 Lemmatizer (†)

The module for the lemmatization is a rule-based system that works by combining the part-of-speech output (Section 3.3) and the results of the morphological analyzer (Section 3.5) so to disambiguate the morphological features using the grammatical annotation. In order to increase the accuracy of the results, the module tries to detect the genre of noun lemmas relying on the analysis of their processed articles. For instance, for the correct lemmatization of "il latte/the milk", the module uses the singular article "il" to identify the correct gender/number of the lemma "latte" and returns "latte/milk" (male, singular) instead of "latta/metal sheet" (female, which plural form is "latte").

In addition, we developed a morphological guesser that is activated whenever a form cannot be linked to any lemma through the morphological analyzer (Section 3.5). Starting from the series form/lemma/pos in the Italian UD datasets, we trained a statistical model using decision trees and probabilities given by frequencies of certain suffixes in the UD. For instance, starting from the non-existent word "insalatando" tagged as verb (probably meaning eating salad), the

guesser starts from the end of the form and, letter by letter, explores the tree of possibilities until it reaches a result with a reasonable accuracy.

The guesser is active by default, but can be deactivated when needed. When active, the guessed lemmas are tagged as such, so that the researcher (or the tool calling Tint) can use this information.

3.7 Verbal tenses classifier

Part-of speech tagger and morphological analyzer released with Tint can identify and classify verbs at token level, but sometimes the modality, form and tense of a verb is the result of a sequence of tokens, as in compound tenses such as participio passato, or passive verb forms. For example, in Italian the word *siamo*, takes as a single token, is the simple present form of the verb *essere*; if we look at the surrounding words, we can have forms such as *siamo andati* (present perfect of verb *andare*, active) or *siamo mangiati* (simple present of verb *mangiare*, passive). For this reason, we include in Tint a tense module to provide a more complete annotation of multi-token verbal forms. The module supports also the analysis of discontinuous expressions, like for example *ho sempre mangiato*.

3.8 Affixes annotator

This module provides a token-level annotation about word derivatives, based on *derIvaTario* [35],[4] a resource manually created to achieve a high accuracy and overcome errors coming from resources developed in a semi-automatic way [42, 36]. The dataset was built segmenting into derivational cycles about 11,000 derivatives and annotating them with a wide array of features. The module uses this resource in input to segment a token into root and affixes, for example *visione* is analysed as *baseLemma=vedere*, *affix=zione* and *allomorph=ione*.

3.9 Multi-word expressions extractor

A specific multi-token annotator has been implemented to recognize more than 13,450 multi-word expressions, the so-called 'polirematiche' [40], manually collected from various online resources. The list includes verbal, nominal, adjectival and prepositional expressions (e.g. *lasciar perdere*, *società per azioni*, *nei confronti di*, *mezzo morto*). This annotator can identify also discontinuous multiwords. For example, in the expression *andare a genio* (Italian phrase that means "to like") an adverb can be included, as in *andare troppo a genio*. Similarly, in such phrases one can find nouns and adjectives (e.g. *lasciare Antonio a piedi*, where *lasciare a piedi* is an Italian multiword for *leave stranded*).

[4] http://derivatario.sns.it/

3.10 Named-entities recognizer (†)

The NER module recognize persons, locations and organizations in the text. It uses a CRF sequence tagger [16] included in Stanford CoreNLP and it is trained on KIND [23], a dataset containing around 340K words taken from Wikinews.

To enhance the classification, Stanford NER also accepts gazettes of names labelled with the corresponding tag. We collect a list of persons, organizations and locations from the Italian Wikipedia using some classes in DBpedia [3]: `Person`, `Organisation`, and `Place`, respectively. In addition to this, we collect the list of streets from OpenStreetMap [22], limiting the extraction to Italian names.

3.11 Temporal expressions extractor and normalizer (*)

Since the first version of Tint, the task of temporal expression extraction was provided as a wrapper to HeidelTime [33], a rule-based state-of-the-art temporal tagger developed at Heidelberg University.

The original English version of CoreNLP uses SUTime [7], a powerful library for processing temporal expressions, built on top of TokensRegex, a framework for defining regular expressions over text and tokens, and mapping matched text to semantic objects. The current version of Tint uses SUTime and a new set of rules written for Italian. It also normalizes the expressions according to the TIMEX3 annotation standard. SUTime is generally run as a subcomponent of the named-entities recognizer annotator (Section 3.10) and is active by default (it can be disabled if not needed).

Recognized temporal expressions can be resolved relative to the document date. For instance, the expression "mercoledì scorso" will be resolved to the Wednesday that is immediately before to the document date, be it the current date or any other date. The document date can be set when Tint is launched, otherwise current date and time are used.

3.12 Constituency parser (*)

A constituency parser is a program that works out the grammatical structure of sentences, for instance, which groups of words go together (as "phrases" and which words are the subject or object of a verb. In Tint this task is performed by shift-reduce [43] parser module included in Stanford CoreNLP.

Data used for training is taken from both the Turin University Treebank [5] and the Parallel TUT [30]. Their licence allows to use it for research purposes.

These treebanks cannot be used as is, because the multiword tokens (see Section 3.4) are denoted by doubling the token. In addition, part-of-speech tags and some constituency labels need to be replaces to make the dataset compatible with the CoreNLP parser. Conversion rules for both tagsets are included in the Tint release. A script to apply the conversion to the dataset is also included.

3.13 Dependency parser (†)

This module provides syntactic analysis of the text and uses a transition-based parser (included in Stanford CoreNLP) which produces typed dependency parses of natural language sentences [8]. The parser is powered by a neural network which accepts word embedding inputs: the model is trained on the UD dataset and the word embeddings are built on the corpus described in Section 3.2.

3.14 Relation extraction

New regulations on transparency and the recent policy for privacy force the public administration (PA) to make their documents available, but also to limit the diffusion of personal data. The relation extraction module represents a first approach to the extraction of sensitive data from PA documents in terms of named entities and semantic relations among them.

For this task, we rely on the Relation Extraction module [34] included in Stanford CoreNLP. For this module to work, a relation must connect two entities. For instance, `address` is used for instance to link a LOC entity representing an address to the person or company to which the address belongs, while `birthDate`, `birthLoc` link respectively the date and location of birth.

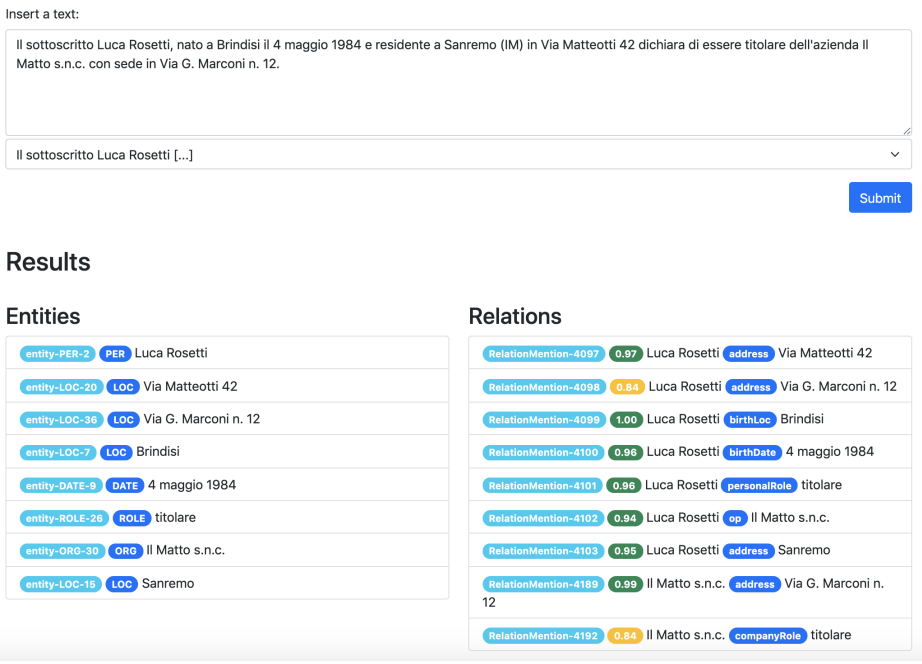

Fig. 2. A screenshot of the relation extraction module online demo.

Some entities are extracted using the named-entities recognizer (Section 3.10) and the temporal expression extractor (Section 3.11). To deal with all the requested relations, some additional entity types are manually added and annotate using the TokenRegexp CoreNLP module (see Section 3.11). Additional entities include, for example, **NUMBER** for numbers (such as VAT), **CF** for the Italian "codice fiscale" sequence of chars, **ROLE** for personal and organisation roles, and so on.

To train the relation extraction module, we use the REDIT dataset, containing documents taken from the PA domain and manually annotated with 19 relations [24].

3.15 Text reuse

Detecting text reuse is useful when, in a document, we want to measure the overlap with a given corpus. This is needed in a number of applications, for example for plagiarism detection, stylometry, authorship attribution, citation analysis, etc. Tint includes a component to deal with this task, i.e. identifying parts of an input text that overlap with a given corpus. First of all, each sentence of the corpus is compared with the sentences in the processed text using the FuzzyWuzzy package[5], a Java fuzzy string matching implementation: this allows the system not to miss expressions that are slightly different with respect to the texts in the original corpus. In this phase, only long spans of text can be considered, as the probability of an incorrect match on fuzzy comparison grows as soon as the text length decreases. A second step checks whether the overlap involves the whole sentence and, if not, it analyzes the two texts and identifies the number of overlapping tokens. Finally, the Stanford CoreNLP quote annotator[6] is used to catch text reuse that is in between quotes, ignoring the length limitation of the fuzzy comparison.

3.16 Readability and corpus statistics

In this module, we compute some metrics that can be useful to assess the readability of a text, partially inspired by [12] and [38]. In particular, we include the following indices:

- Number of content words, hyphens (using iText Java Library[7]), sentences having less than a fixed number of words, distribution of tokens based on part-of-speech.
- Type-token ratio (TTR), i.e. the ratio between the number of different lemmas and the number of tokens; high TTR indicates a high degree of lexical variation.
- Lexical density, i.e. the number of content words divided by the total number of words.

[5] https://github.com/xdrop/fuzzywuzzy
[6] https://stanfordnlp.github.io/CoreNLP/quote.html
[7] https://github.com/itext/itextpdf

- Amount of coordinate and subordinate clauses, along with the ratio between them.
- Depth of the parse tree for each sentence: both average and max depth are calculated on the whole text.
- Gulpease formula [19] to measure the readability at document level.
- Text difficulty based on word lists from De Mauro's Dictionary of Basic Italian[8].

In addition, a set of extractors described in [31] and mainly defining the neo-standard Italian used by high school students (such as anglicisms, euphonic "D", and much more) are available out-of-the-box in Tint.

Finally, a collection of CoreNLP annotators have been developed to extract statistics that can be used, for instance, to analyse traits of interest in texts. More specifically, the provided modules can mark and compute words and sentences based on token, lemma, part-of-speech and word position in the sentence.

3.17 Keywords extraction

Keyword extraction in Tint is performed by Keyphrase Digger[9] [21], a rule-based system for keyphrase extraction. It combines statistical measures with linguistic information given by part-of-speech patterns to identify and extract weighted keyphrases from texts.

3.18 Entity linking

The entity linking task consists in disambiguating a word (or a set of words) and link them to a knowledge base (KB). The biggest (and most used) available KB is Wikipedia, and almost every linking tool relies on it. The Tint pipeline provides a wrapper annotator that can connect to DBpedia Spotlight[10] [10] and The Wiki Machine[11] [2]. Both tools are distributed as open source software and can be used by the annotator both as external services or through a local installation.

4 Evaluation

Tint is a complex system that relies on a variety of modules interacting each other. For this reason, the accuracy on the single tasks does not reach state-of-the-art accuracy. Nevertheless, Tint performs as a reasonable level in all tasks it performs.

An accurate evaluation of most of its modules (especially the ones that use machine learning techniques), with a comparison with other NLP Italian tools, is available in the previous papers [26, 27, 24].

[8] http://bit.ly/nuovo-demauro
[9] https://dh.fbk.eu/2015/12/kd-keyphrase-digger/
[10] https://www.dbpedia-spotlight.org/
[11] https://bitbucket.org/fbk/airpedia/wiki/Tutorial

5 Tint distibution

The Tint pipeline is released as an open source software under the GNU General Public License (GPL), version 3. It can be download from the Tint website[12] as a standalone package, or it can be integrated into an existing application as a Maven dependency. The source code is available on Github.[13]

The tool is written using the Stanford CoreNLP paradigm, therefore a third part software can be integrated easily into the pipeline.

Fig. 3. A screenshot of the Tintful web interface.

Along with Tint, one can also try Tintful[14] [17], a NLP annotation software that can be used both to manually annotate texts and to fix mistakes in NLP pipelines (and, in particular, in Tint). Using a paradigm similar to wiki-like systems, a user who notices some wrong annotation can easily fix it and submit the resulting (and right) entry back to the tool developers. The Tint online demo, linked from the project website, uses Tintful as graphical interface and is configured to show most of the modules described in this paper. Therefore the annotation provided by the modules working on machine learning algorithms that need to be trained over annotated data (named-entity recognizer, part-of-speech tagger, dependency parser) can be edited by the occasional user. The resulting annotation will be manually checked by linguists and added to the next training session. Figure 3 shows the web interface of Tintful.

[12] http://tint.fbk.eu/
[13] https://github.com/dhfbk/tint/
[14] https://github.com/dhfbk/tintful

6 Conclusions and Future Work

In this paper, we presented the last release of Tint, a simple, fast and accurate NLP pipeline for Italian, based on Stanford CoreNLP. In the new version, we fixed some bugs and improved some of the existing modules. We also added a set of components for fine-grained linguistics analysis that were not available so far.

In the future, we plan to improve the suite and extend it with additional modules, in particular Word Sense Disambiguation (WSD) based on linguistic resources such as MultiWordNet [28] and Semantic Role Labelling, by porting to Italian resources such as FrameNet [4], now available only in English.

We also plan to increase the accuracy of the trained modules (such as part-of-speech tagger and named-entity recognizer) using deep learning techniques and including a pretrained language model at a different granularity (words, characters) into the process [1].

References

1. Akbik, A., Blythe, D., Vollgraf, R.: Contextual string embeddings for sequence labeling. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 1638–1649. Association for Computational Linguistics, Santa Fe, New Mexico, USA (Aug 2018)
2. Aprosio, A.P., Giuliano, C.: The wiki machine: an open source software for entity linking and enrichment. ArXiv e-prints (2016)
3. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. In: Aberer, K., Choi, K.S., Noy, N., Allemang, D., Lee, K.I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) The Semantic Web. pp. 722–735. Springer Berlin Heidelberg, Berlin, Heidelberg (2007)
4. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The berkeley framenet project. In: Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 1. pp. 86–90. Association for Computational Linguistics (1998)
5. Bosco, C., Lombardo, V., Vassallo, D., Lesmo, L.: Building a treebank for Italian: a data-driven annotation schema. In: Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00). European Language Resources Association (ELRA), Athens, Greece (May 2000), http://www.lrec-conf.org/proceedings/lrec2000/pdf/220.pdf
6. Bosco, C., Montemagni, S., Simi, M.: Converting Italian treebanks: Towards an Italian Stanford dependency treebank. In: Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse. pp. 61–69. Association for Computational Linguistics, Sofia, Bulgaria (Aug 2013), https://aclanthology.org/W13-2308
7. Chang, A.X., Manning, C.: SUTime: A library for recognizing and normalizing time expressions. In: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12). pp. 3735–3740. European Language Resources Association (ELRA), Istanbul, Turkey (May 2012)

8. Chen, D., Manning, C.D.: A fast and accurate dependency parser using neural networks. In: EMNLP. pp. 740–750 (2014)
9. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: Gate: An architecture for development of robust hlt applications. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics. pp. 168–175. ACL '02, Association for Computational Linguistics, Stroudsburg, PA, USA (2002). https://doi.org/10.3115/1073083.1073112, http://dx.doi.org/10.3115/1073083.1073112
10. Daiber, J., Jakob, M., Hokamp, C., Mendes, P.N.: Improving efficiency and accuracy in multilingual entity extraction. In: Proceedings of the 9th International Conference on Semantic Systems (I-Semantics) (2013)
11. De La Briandais, R.: File searching using variable length keys. In: Papers Presented at the the March 3-5, 1959, Western Joint Computer Conference. pp. 295–298. IRE-AIEE-ACM '59 (Western), ACM, New York, NY, USA (1959). https://doi.org/10.1145/1457838.1457895, http://doi.acm.org/10.1145/1457838.1457895
12. Dell'Orletta, F., Montemagni, S., Venturi, G.: Read-it: Assessing readability of italian texts with a view to text simplification. In: Proceedings of the Second Workshop on Speech and Language Processing for Assistive Technologies. pp. 73–83. SLPAT '11, Association for Computational Linguistics, Stroudsburg, PA, USA (2011), http://dl.acm.org/citation.cfm?id=2140499.2140511
13. Dell'Orletta, F., Venturi, G., Cimino, A., Montemagni, S.: T2kˆ 2: a system for automatically extracting and organizing knowledge from texts. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014) (2014)
14. Emanuele Pianta, C.G., Zanoli, R.: The textpro tool suite. In: Chair), N.C.C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Tapias, D. (eds.) Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08). European Language Resources Association (ELRA), Marrakech, Morocco (2008)
15. Ferrucci, D., Lally, A.: Uima: An architectural approach to unstructured information processing in the corporate research environment. Nat. Lang. Eng. **10**(3-4), 327–348 (Sep 2004). https://doi.org/10.1017/S1351324904003523, http://dx.doi.org/10.1017/S1351324904003523
16. Finkel, J.R., Grenager, T., Manning, C.: Incorporating non-local information into information extraction systems by Gibbs sampling. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05). pp. 363–370. Association for Computational Linguistics, Ann Arbor, Michigan (Jun 2005). https://doi.org/10.3115/1219840.1219885, https://aclanthology.org/P05-1045
17. Frasnelli, V., Bocchi, L., Palmero Aprosio, A.: Erase and rewind: Manual correction of NLP output through a web interface. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations. pp. 107–113. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.acl-demo.13, https://aclanthology.org/2021.acl-demo.13
18. Honnibal, M., Johnson, M.: An improved non-monotonic transition system for dependency parsing. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1373–1378. Asso-

ciation for Computational Linguistics, Lisbon, Portugal (September 2015), https://aclweb.org/anthology/D/D15/D15-1162
19. Lucisano, P., Piemontese, M.E.: GULPEASE: una formula per la predizione della difficoltà dei testi in lingua italiana. Scuola e città **3**(31), 110–124 (1988)
20. de Marneffe, M.C., Manning, C.D., Nivre, J., Zeman, D.: Universal Dependencies. Computational Linguistics **47**(2), 255–308 (07 2021)
21. Moretti, G., Sprugnoli, R., Tonelli, S.: Digging in the dirt: Extracting keyphrases from texts with kd. CLiC it p. 198 (2015)
22. OpenStreetMap contributors: Planet dump retrieved from https://planet.osm.org . https://www.openstreetmap.org (2017)
23. Paccosi, T., Palmero Aprosio, A.: KIND: an Italian Multi-Domain Dataset for Named Entity Recognition. In: arXiv preprint (2021)
24. Paccosi, T., Palmero Aprosio, A.: REDIT: a Tool and Dataset for Extraction of Personal Data in Documents of the Public Administration Domain. In: CLiC-it 2021 Italian Conference on Computational Linguistics (2021)
25. Padró, L., Stanilovsky, E.: Freeling 3.0: Towards wider multilinguality. In: LREC2012 (2012)
26. Palmero Aprosio, A., Moretti, G.: Italy goes to Stanford: a collection of CoreNLP modules for Italian. ArXiv e-prints (Sep 2016)
27. Palmero Aprosio, A., Moretti, G.: Tint 2.0: an all-inclusive suite for nlp in italian. In: Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it. vol. 10, p. 12 (2018)
28. Pianta, E., Bentivogli, L., Girardi, C.: Developing an aligned multilingual database. In: Proc. 1st Int'l Conference on Global WordNet. Citeseer (2002)
29. Qi, P., Zhang, Y., Zhang, Y., Bolton, J., Manning, C.D.: Stanza: A Python natural language processing toolkit for many human languages. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (2020), https://nlp.stanford.edu/pubs/qi2020stanza.pdf
30. Sanguinetti, M., Bosco, C.: PartTUT: The Turin University Parallel Treebank, pp. 51–69. Springer International Publishing, Cham (2015)
31. Sprugnoli, R., Tonelli, S., Aprosio, A.P., Moretti, G.: Analysing the evolution of students' writing skills and the impact of neo-standard italian with the help of computational linguistics. In: Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2018). Torino, Italy (2018)
32. Straka, M., Straková, J.: Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In: Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. pp. 88–99. Association for Computational Linguistics, Vancouver, Canada (2017)
33. Strötgen, J., Armiti, A., Van Canh, T., Zell, J., Gertz, M.: Time for more languages: Temporal tagging of arabic, italian, spanish, and vietnamese. ACM Transactions on Asian Language Information Processing (TALIP) **13**(1), 1–21 (2014)
34. Surdeanu, M., McClosky, D., Smith, M., Gusev, A., Manning, C.: Customizing an information extraction system to a new domain. In: Proceedings of the ACL 2011 Workshop on Relational Models of Semantics. pp. 2–10. Association for Computational Linguistics, Portland, Oregon, USA (Jun 2011), https://aclanthology.org/W11-0902
35. Talamo, L., Celata, C., Bertinetto, P.M.: DerIvaTario: An annotated lexicon of Italian derivatives. Word Structure **9**(1), 72–102 (2016)
36. Tamburini, F., Melandri, M.: Anita: a powerful morphological analyser for italian. In: Chair), N.C.C., Choukri, K., Declerck, T., Doğan, M.U., Maegaard, B.,

Mariani, J., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey (2012)
37. Tonelli, S., Palmero Aprosio, A., Mazzon, M.: The impact of phrases on italian lexical simplification. In: Proceedings of the Fourth Italian Conference on Computational Linguistics (CLiC-it 2017). pp. 316–320 (2017)
38. Tonelli, S., Tran Manh, K., Pianta, E.: Making readability indices readable. In: Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations. pp. 40–48. Association for Computational Linguistics, Montréal, Canada (June 2012), http://www.aclweb.org/anthology/W12-2206
39. Toutanova, K., Klein, D., Manning, C.D., Singer, Y.: Feature-rich part-of-speech tagging with a cyclic dependency network. In: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1. pp. 173–180. NAACL '03, Association for Computational Linguistics, Stroudsburg, PA, USA (2003). https://doi.org/10.3115/1073445.1073478, http://dx.doi.org/10.3115/1073445.1073478
40. Voghera, M.: Polirematiche. La formazione delle parole in italiano pp. 56–69 (2004)
41. Zanchetta, E., Baroni, M.: Morph-it! A free corpus-based morphological resource for the Italian language. Corpus Linguistics 2005 **1**(1) (2005)
42. Zanchetta, E., Baroni, M.: Morph-it! a free corpus-based morphological resource for the italian language. In: Proceedings of corpus linguistics 2005. University of Birmingham UK (2005)
43. Zhu, M., Zhang, Y., Chen, W., Zhang, M., Zhu, J.: Fast and accurate shift-reduce constituent parsing. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 434–443. Association for Computational Linguistics, Sofia, Bulgaria (Aug 2013), https://aclanthology.org/P13-1043