Italy goes to Stanford: a collection of CoreNLP modules for Italian

Alessio Palmero Aprosio
Fondazione Bruno Kessler
Trento, Italy
aprosio@fbk.eu

Giovanni Moretti
Fondazione Bruno Kessler
Trento, Italy
moretti@fbk.eu

Abstract

English. In this we paper present Tint, an easy-to-use set of fast, accurate and extendable Natural Language Processing modules for Italian. It is based on Stanford CoreNLP and is freely available as a standalone software or a library that can be integrated in an existing project.

Italiano. *In questo articolo presentiamo Tint, una collezione di moduli semplici, veloci e personalizzabili per l'analisi di testi in Italiano. Tint è basato su Stanford CoreNLP e può essere scaricato gratuitamente come software stand-alone o come libreria da integrare in progetti esistenti.*

1 Introduction

In recent years, the Natural Language Processing (NLP) technologies have become a fundamental basis for complex tasks, such as Question Answering, Event Identification and Topic Classification. While most of the NLP tools freely available on the web (such as Stanford CoreNLP[1] and OpenNLP[2]) are designed for English and sometimes adapted to other languages, there is a lack of this kind of resources for Italian.

In this paper, we present Tint, a suite of ready-to-use modules for NLP that is:

New. Tint is the first completely free and open source tool for NLP in Italian.

Simple. Tint can be downloaded and used out-of-the-box (see Section 5). In addition, it relies on Stanford CoreNLP Java interface, therefore it can be included easily into an existing project.

Modular. Tint can be extended using the CoreNLP Java interfaces. At the same time, existing modules can be replaced with more customized ones.

Efficient. In its default configuration, Tint is faster than most of its competitors (see Section 4).

Accurate. Most of Tint modules have a state-of-the-art accuracy (see Section 4).

Free. Tint is released as open source software under GNU GPL.

2 Architecture

The Tint pipeline is based on Stanford CoreNLP (Manning et al., 2014), an open-source framework written in Java, that provide most of the commons Natural Language Proccessing tasks out-of-the-box in various language. The framework provides also an easy interface to extend the annotation to new tasks and/or languages. Differently from some similar tools, such as UIMA (Ferrucci and Lally, 2004) and GATE (Cunningham et al., 2002), CoreNLP is easy to use and does not require it to be learnt: a basic object-oriented programming skill is enough. In Tint, we use this framework to both port the most common NLP tasks to Italian and add some new annotators for external tools, such as entity linking, temporal expression identification, keyword extraction.

3 Modules

3.1 Tokenizer

This module provides text segmentation in tokens and sentences. At first, the text is grossly tokenized; 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, dates).

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

3.2 Morphological Analyzer

The morphological analyzer module provides the full list of morphological features for each annotated token. The current version of this module has been trained with the Morph-it lexicon (Zanchetta and Baroni, 2005), but it's possible to extend or retrain it with other Italian datasets. In order to grant fast performance, the model storage has been implemented with the mapDB Java library[3] that provides an excellent variation of the Cassandra's Sorted String Table. To extend the coverage of the results, especially for the complex forms, such as "porta-ce-ne", "portar-glie-lo" or "bi-direzionale", the module tries to decompose the token into prefix-root-infix-suffix and attempts to resolve the root form.

3.3 Part-of-speech tagger

The part-of-speech annotation is provided through the Maximum Entropy implementation (Toutanova et al., 2003) included in Stanford CoreNLP. The model is trained on the Universal Dependencies[4] (UD) dataset for Italian (Bosco et al., 2013), a dataset – freely available for research purpose – containing more than 300K tokens annotated with lemma, part-of-speech and syntactic dependencies. As an alternative, a wrapper annotator that uses TreeTagger is also available in Tint.

3.4 Lemmatizer

The module for the lemmatization is a rule-based system that works by combining the Part-of-Speech output and the results of the Morphological Analyzer 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 to 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").

3.5 Named Entity Recognition and Classification

The NER module recognize persons, locations and organizations in the text. It uses a CRF sequence tagger (Finkel et al., 2005) included in Stanford CoreNLP and it is trained on the I-CAB (Magnini et al., 2006), a dataset containing 180K words taken from the Italian newspaper "L'Adige".

3.6 Dependency Parsing

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 (Chen and Manning, 2014). The parser is powered by a neural network which accepts word embedding inputs: the model is trained on the UD dataset (see Section 3.3) and the word embeddings are built on the Paisà corpus (Lyding et al., 2014), that contains 250M tokens of freely available and distributable texts harvested from the web.

3.7 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[5] (Daiber et al., 2013) and The Wiki Machine[6] (Giuliano et al., 2009). Both tools are distributed as open source software and can be used by the annotator both as external services or through a local installation.

3.8 Temporal Expression Extraction and Normalization

The task of temporal expression extraction is included in Tint as a wrapper to HeidelTime (Strötgen and Gertz, 2013), a rule-based state-of-the-art temporal tagger developed at Heidelberg University. HeidelTime also normalizes the expressions according to the TIMEX3 annotation standard. The software is released under the GPL license, therefore it can be used both for educational and commercial purposes.

[3] http://www.mapdb.org
[4] http://universaldependencies.org/
[5] http://bit.ly/dbpspotlight
[6] http://bit.ly/thewikimachine

3.9 Keyword extraction

Keyword extraction in Tint is performed by Keyphrase Digger (Moretti et al., 2015), 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. The CoreNLP annotator for Keyphrase Digger is included in the Tint pipeline, but the main software must be downloaded and installed from the official website[7] as it is not released open source.

4 Evaluation

Tint includes a rich set of tools, evaluated separately. In some cases, an evaluation based on the accuracy is not possible, because of the lack of available gold standard or because the tool outcome is not comparable to other tools' ones.

When possible, Tint is compared with existing pipelines that work with the Italian language: Tanl (Attardi et al., 2010), TextPro (Pianta et al., 2008) and TreeTagger (Schmid, 1994).

In calculating speed, we run each experiment 10 times and consider the average execution time. When available, multi-thread capabilities have been disabled. All experiments have been executed on a 2,3 GHz Intel Core i7 with 16 GB of memory.

The Tanl API is not available as a downloadable package, but it's only usable online through a REST API, therefore the speed may be influenced by the network connection. In addition, the Tanl API does not provide offsets for the annotated text, nor it allows a text to be uploaded already tokenized and divided in sentences, therefore an automatic alignment was needed. The tools used for this alignment are distributed as part of the Tint software.

No evaluation is performed for the Tint annotators that act as wrappers for an external tools (temporal expression tagging, entity linking, keyword extraction).

4.1 Tokenization and sentence splitting

For the task of tokenization and sentence splitting, Tint outperforms in speed both TextPro and Tanl (see Table 1). The number of tokens per second can be further increased by tuning the features (for example, by deactivating the regular expressions that recognize e-mail or web addresses).

[7] `http://dh.fbk.eu/technologies/kd`

System	Speed (tok/sec)
Tint	**80,000**
Tanl API	30,000
TextPro 2.0	35,000

Table 1: Tokenization and sentence splitting speed.

4.2 Part-of-speech tagging

The evaluation of the part-of-speech tagging is performed against the test set included in the UD dataset, containing 10K tokens. As the tagset used is different for different tools, the accuracy is calculated only on five coarse-grained types: nouns (N), verbs (V), adverbs (B), adjectives (A) and other (O). For each tool, the corresponding tagset is converted to this tagset and accuracy is calculated dividing the number of times the tagger gets the right answer by the total number of tags in the dataset. Table 2 shows the results.

System	Speed (tok/sec)	Accuracy
Tint	28,000	**98%**
Tanl API	20,000	n.a.
TextPro 2.0	20,000	96%
TreeTagger	**190,000**[8]	92%

Table 2: Evaluation of part-of-speech tagging.

4.3 Lemmatization

Like part-of-speech tagging, lemmatization is evaluated, both in terms of accuracy and execution time, on the UD test set. When the lemma is guessed starting form a morphological analysis (such as in Tint and TextPro), the speed is calculated by including both tasks. Table 3 shows the results. All the tools reach the same accuracy of 96% (with minor differences that are not statistically significant).

System	Speed (tok/sec)	Accuracy
Tint	97,000	96%
TextPro 2.0	9,000	96%
TreeTagger	**190,000**[8]	96%

Table 3: Evaluation of lemmatization.

4.4 Named Entities Recognition

For Named Entity Recognition, we evaluate and compare our system with the test set available on the I-CAB dataset. We consider three classes:

[8] The (considerable) speed of TreeTagger includes both lemmatization and part-of-speech tagging.

PER, ORG, LOC. Both Tanl and TextPro deal also with the GPE class, but we merged it to LOC, as it has been done during the training of Tint. We needed to retrain the EntityPro module of TextPro from scratch (with three classes), as the original model already contains the I-CAB test set, therefore it would overfit the results. In training Tint, we add some gazette of names, to help the classifier to recognize entities that are not present in the training set. In particular, we extracted a list of persons, locations and organizations by querying the Airpedia database (Palmero Aprosio et al., 2013) for Wikipedia pages classified as `Person`, `Place` and `Organisation`, respectively. The whole data used for training the NER is available for download from the Tint website. Table 4 shows the results of the named entity recognition task.

System	Speed	P	R	F_1
Tint	**30,000**	**84.37**	79.97	**82.11**
TextPro 2.0	4,000	81.78	**80.78**	81.28
Tanl API	16,000	72.89	52.50	61.04

Table 4: Evaluation of the NER.

4.5 Dependency parsing

The evaluation of the dependency parser is performed against Tanl and TextPro w.r.t the usual metrics Labeled Attachment Score (LAS) and Unlabeled Attachment Score (UAS). While Tint is trained on the UD dataset, the parsers included in Tanl (Attardi et al., 2013) and TextPro (Lavelli, 2013) use part of the Turin University Treebank (TUT) (Bosco et al., 2000), as released for the Evalita 2011 parsing task (Magnini et al., 2013). For this reason, the comparison between the two system is not completely fair: on the one hand, the TUT dataset is smaller than the UD; on the other hand, the UD is an automatic combination of two different treebanks, that have been annotated using different guidelines (Bosco et al., 2013). Table 5 shows the results: the Tint evaluation has been performed on the UD test data; LAS and UAS for TextPro and Tanl is taken directly from the Evalita 2011 proceedings.

System	Speed	LAS	UAS
Tint	**9,000**	84.67	87.05
TextPro 2.0	1,300	87.30	91.47
Tanl (DeSR)	900	**89.88**	**93.73**

Table 5: Evaluation of the dependency parsing.

5 The tool

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[9] as a standalone package, or it can be integrated into an existing application as a Maven dependency.

The tool is written using the Stanford CoreNLP paradigm, therefore a third part software can be integrated easily into the pipeline. Tint accepts plain text or Newsreader Annotation Format (NAF) (Fokkens et al., 2014) as input, and CoNLL or NAF as output.

6 Conclusion and Future Work

In this paper we presented Tint, a simple, fast and accurate NLP pipeline for Italian, based on Stanford CoreNLP. Currently, we offer out-of-the-box NLP annotation for part-of-speech, lemma, named entities, links to Wikipedia, dependency parsing, time expression identification and keyword extraction; additional custom modules can be added and replaced easily by implementing the CoreNLP Java interfaces.

In the future, we plan to better tune the various modules that rely on machine learning (such as dependency parsing, part-of-speech tagging and named entity recognition), that in this preliminary version of Tint have been trained without any linguistic optimization.

We are currently working on new modules, in particular Word Sense Disambiguation (WSD) w.r.t. linguistic resources such as MultiWordNet (Pianta et al., 2002) and Semantic Role Labelling, by porting to Italian resources such as Framenet (Baker et al., 1998), now available in English.

On the technical side, we are updating some modules to work multi-thread. The Tint pipeline will also be integrated into PIKES (Corcoglioniti et al., 2016), a tool that extracts knowledge from texts using NLP annotation and outputs it in a queryable form (such RDF triples).

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).

[9] `http://tint.fbk.eu/`

References

G. Attardi, S. Dei Rossi, and M. Simi. 2010. The Tanl Pipeline. In *Proc. of LREC Workshop on WSPP*.

Giuseppe Attardi, Maria Simi, and Andrea Zanelli. 2013. Tuning desr for dependency parsing of italian. In *Evaluation of Natural Language and Speech Tools for Italian*, pages 37–45. Springer.

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. 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*, pages 86–90. Association for Computational Linguistics.

Cristina Bosco, Vincenzo Lombardo, Daniela Vassallo, and Leonardo Lesmo. 2000. Building a treebank for italian: a data-driven annotation schema. In *LREC*. Citeseer.

Cristina Bosco, Simonetta Montemagni, and Maria Simi. 2013. Converting italian treebanks: Towards an italian stanford dependency treebank.

Danqi Chen and Christopher D Manning. 2014. A fast and accurate dependency parser using neural networks. In *EMNLP*, pages 740–750.

Francesco Corcoglioniti, Marco Rospocher, and Alessio Palmero Aprosio. 2016. A 2-phase frame-based knowledge extraction framework. In *Proc. of ACM Symposium on Applied Computing (SAC'16)*.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, and Valentin Tablan. 2002. Gate: An architecture for development of robust hlt applications. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 168–175, Stroudsburg, PA, USA. Association for Computational Linguistics.

Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N. Mendes. 2013. Improving efficiency and accuracy in multilingual entity extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*.

David Ferrucci and Adam Lally. 2004. Uima: An architectural approach to unstructured information processing in the corporate research environment. *Nat. Lang. Eng.*, 10(3-4):327–348, September.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 363–370, Stroudsburg, PA, USA. Association for Computational Linguistics.

Antske Fokkens, Aitor Soroa, Zuhaitz Beloki, Niels Ockeloen, German Rigau, Willem Robert van Hage, and Piek Vossen. 2014. Naf and gaf: Linking linguistic annotations. In *Proceedings 10th Joint ISO-ACL SIGSEM Workshop on Interoperable Semantic Annotation*, pages 9–16.

Claudio Giuliano, Alfio Massimiliano Gliozzo, and Carlo Strapparava. 2009. Kernel methods for minimally supervised wsd. *Comput. Linguist.*, 35(4):513–528, December.

Alberto Lavelli. 2013. An ensemble model for the evalita 2011 dependency parsing task. In *Evaluation of Natural Language and Speech Tools for Italian*, pages 30–36. Springer.

Verena Lyding, Egon Stemle, Claudia Borghetti, Marco Brunello, Sara Castagnoli, Felice Dell'Orletta, Henrik Dittmann, Alessandro Lenci, and Vito Pirrelli. 2014. The paisa corpus of italian web texts. In *Proceedings of the 9th Web as Corpus Workshop (WaC-9)*, pages 36–43.

Bernardo Magnini, Emanuele Pianta, Christian Girardi, Matteo Negri, Lorenza Romano, Manuela Speranza, Valentina Bartalesi Lenzi, and Rachele Sprugnoli. 2006. I-cab: the italian content annotation bank. In *Proceedings of LREC*, pages 963–968. Citeseer.

Bernardo Magnini, Francesco Cutugno, Mauro Falcone, and Emanuele Pianta. 2013. *Evaluation of Natural Language and Speech Tool for Italian: International Workshop, EVALITA 2011, Rome, January 24-25, 2012, Revised Selected Papers*, volume 7689. Springer.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60.

Giovanni Moretti, Rachele Sprugnoli, and Sara Tonelli. 2015. Digging in the dirt: Extracting keyphrases from texts with kd. *CLiC it*, page 198.

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*.

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. Developing an aligned multilingual database. In *Proc. 1st Int'l Conference on Global WordNet*. Citeseer.

Emanuele Pianta, Christian Girardi, and Roberto Zanoli. 2008. The textpro tool suite. In *LREC*. Citeseer.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. 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*, NAACL '03, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

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