

# SenTag 2.0: A Cooperative Annotation Tool

Alberto Zerbinati<sup>1</sup>, Andrea Loreggia<sup>2,\*</sup>, Giuseppe Contissa<sup>3</sup>, Federico Galli<sup>3</sup>,  
Francesca Lagioia<sup>3,4</sup> and Giovanni Sartor<sup>3,4</sup>

<sup>1</sup>University of Padova, Italy

<sup>2</sup>University of Brescia, Italy

<sup>3</sup>CIRSFID Alma AI - University of Bologna, Italy

<sup>4</sup>European University Institute, Italy

## Abstract

In this work, we present SenTag 2.0, a web-based application for the semantic annotation of textual documents. SenTag provides an easy-to-use user interface through which multiple users can annotate a corpus of documents and produce XML files that can be validated against a schema. SenTag 2.0 aims at assisting the user during the tagging process and reducing errors in the output documents.

## Keywords

Legal Analytics, Annotation Web-Interface, XML schema

## 1. Introduction

In some areas of NLP, data is collected by building large corpora of annotated documents [1]. To annotate the documents, researchers employ groups of domain experts, which are tasked with identifying and describing the relevant components in the text. The resulting corpus of annotated documents is then used to train artificial intelligence models [2]. To identify and describe entities, arguments, and other information in the documents, domain experts usually adopt a predefined structured language [3], such as Extensible Markup Language (XML). This allows the original document to be enriched with a semantic annotation that identifies important information included in the text [1]. Here, we present SenTag 2.0, a web-based application meant to provide a user-friendly interface that facilitates the annotation process. It enables annotators to interact with documents without dealing directly with the used markup language. Besides the easy-to-use tagging interface, the main contribution of the present version— which differentiates SenTag 2.0 from other solutions— is the integration of a separate visualization of the annotation that allows interaction with a tagged document through graphs that allow specifying relationships among arguments or other entities. We also introduce the possibility to import XML files into the application in order to visualize and continue the annotation process using SenTag 2.0. The full code is publicly available at <https://github.com/AlbertoZerbinati/sentag>.

*ISWC-Posters-Demos-Industry 2022 (International Semantic Web Conference (ISWC) 2022: Posters, Demos, and Industry Tracks)*

\*Corresponding author.

✉ [alberto.zerbinati@studenti.unipd.it](mailto:alberto.zerbinati@studenti.unipd.it) (A. Zerbinati); [andrea.loreggia@gmail.com](mailto:andrea.loreggia@gmail.com) (A. Loreggia)

ORCID [0000-0002-9846-0157](https://orcid.org/0000-0002-9846-0157) (A. Loreggia)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

## 2. Related Work

We report a non-exhaustive overview of recent works. SLATE [4] is a lightweight annotation tool that supports annotation at different scales (spans of characters, tokens, and lines, or a document) and of different types (free text, labels, and links). YEDDA [5] is a desktop application that supports annotation of character spans with up to 8 categories using a GUI. It is designed to be lightweight, with no external dependencies, though it only supports Python 2. INCEpTION [6] is a semantic annotation platform developed under the framework of the INCEpTION project. The annotation process provides labels on each annotation, indicating the specific elements, attributes, etc., to make their recognition easier. WebAnno is a generic web-based annotation tool for distributed teams [7]. HUMAN [8] is a web-based annotation tool that covers several annotation tasks on both textual and image data, and adopts an internal deterministic state machine that enables researchers to chain different annotation tasks in an interdependent manner. EntiTies [9] is another web application that allows users to annotate a text from scratch, to process it automatically, or to process it automatically and then correct it.

## 3. SenTag

Although being domain-agnostic, SenTag 2.0 is inspired by the efforts spent in the field of legal analytics within the ADELE Project (Analytics for DEcision of LEgal cases)<sup>1</sup>. The tool was used and tested by a team of legal experts to annotate judicial decisions, with excellent feedback in terms of design, usability and functionality. SenTag 2.0 [10] has been developed in Python 3.9 as a Django 3.2 app employing VUE 3 for the graphical interface. The application is based on NER Annotator, from which it derives and expands the tagging part. We developed the application to implement the following requirements: (i) security and user management; (ii) multiple annotators and agreement score; (iii) intuitive interface for text annotation; (iv) graph implementation for entities and relations. The main goal of the SenTag project is to support domain-experts building manually annotated datasets of documents. Thanks to SenTag, domain experts can annotate documents without any knowledge of the underlining markup language. A Javascript module allows for a client-side implementation of the annotation, avoiding useless transfers of information to the server. The information is sent to the server only when a user saves the changes. All these modules interact with each other and store information in a SQLite database. The internal representation of information is different from the ways it is presented to users, according to the standard Model-View-Controller (MVC) design-pattern [11]. In the following, we shall describe how we implement the requirements listed above.

**Security and User Management.** Being a web-based application, it is necessary to protect documents from unauthorized access. It is also important to assign each authenticated user the correct level of rights and their specific role in the process. There are three available groups levelled in a hierarchical structure. Users in a group have the right to do all tasks allowed to lower levels. These groups are: (i) **Admin**: this is the group with the highest level of rights. Admin users can create, delete and update all users, and assign any rights. They can also assign

---

<sup>1</sup>ADELE (Analytics for DEcision of LEgal cases) Funded by the European Union's Justice Programme (Grant Agreement n. 101007420), information available at <https://site.unibo.it/adele/en/project>.

users to a specific group; (ii) **Editor**: this is the group with middle-level rights. Editor users are in charge of: (i) uploading texts and XML schemas; (ii) grouping documents into tasks; (iii) assigning tasks to schemas; (iv) assigning annotators to tasks they should perform; (v) checking for annotators agreement; (vi) downloading annotated documents; (iii) **Annotator**: this is the group with the lowest level of rights. Annotator users can only annotate the documents assigned to them, using the set of tags specified in the schema associated with the task. Each annotator can validate his or her work using the associated XML schema in order to check for annotation errors or flaws; the annotation is enriched with graphs that can be used by annotators to establish relationships among entities and/or arguments identified in the document.

**Multiple Annotators and Agreement Score.** Documents are assigned to one or multiple tasks that should be accomplished by annotators. Thus a task corresponds to a set of documents, a set of users assigned to that task, and a schema. Admins and editors can check the annotation status of a task through the web-interface. It allows seeing some statistics about the quality and the validity of the annotation performed on the document. In particular, it reports the Krippendorff's alpha [12] score for each document. This is a metric adopted to describe the level of agreement for a group of annotators working on the same document. The score quantifies how similarly annotators tagged the same text. Moreover, for each annotator, the interface reports whether he or she has completed the annotation for a document. It also reports whether each annotated document passed the validity check against the XML schema. Editors are entitled to download the resulting XML files. This can be done at different levels of granularity, in particular, they can download: i) the XMLs resulting from the annotation of a single annotator on a single document; ii) the XMLs resulting from the annotation of all the annotators on a single document; iii) the XMLs resulting from the annotation of a single annotator on all the documents; iv) all the XMLs resulting from the annotation of all annotators on all documents.

**Text Annotation.** Annotation tags are meant to characterise important information in the text. In order to do that, Extensible Markup Language (XML) schemas are employed to define tags for the relevant entities that can be encountered in the documents and their hierarchy. Each entity is indeed identified by a tag that is used to surround the corresponding text. Moreover, each tag can be enriched with a set of attributes that describe possible relationships among entities or augment the semantics of the tag. Thus, the XML schema specifying tags and attributes for a certain document corpus is designed and defined a priori and associated with each document (via the task). The schema specifies what tags and what attributes each user can insert into the document. It can be used also to validate the output of the annotation phase. Annotator users are in charge of this process. After being authenticated, each annotator can see the set of tasks that are assigned to them and after selecting the task, they can see all the documents that they should annotate.

**Graphs of Arguments and Relationships.** During the tagging phase, it is possible to visualise some of the entities identified in the text as nodes in a graph. Presently, the platform allows two different graphs to be built (the graph of arguments and the graph of relationships), but this feature can be easily extended to other types of graphs. A node in either graph appears as soon as an annotator tags a part of the text with a specific tag. The XML schema is used to specify what tags represent arguments and which ones represent entities. This is done by adding the attribute *GRAPH* or the attribute *RELATIONS* in the schema. Graph visualisation

is useful during the tagging process as it allows to establish relationships between entities in an intuitive way.

## Acknowledgments

This work has been supported by the H2020 ERC Project “CompuLaw” (G.A. 833647); the ADELE (Analytics for DEcision of LEgal Cases) project under the European Union’s Justice programme (G.A. 101007420), and by the LAILA (Legal Analytics for Italian Law) project under the Italian Ministry of Education and Research’s PRIN programme (G.A. 2017NCPZ22).

## References

- [1] P. Poudyal, J. Savelka, A. Ieven, et al., Echr: Legal corpus for argument mining, in: Proceedings of the 7th Workshop on Argument Mining, 2020, pp. 67–75.
- [2] M. Lippi, P. Torroni, Argumentation mining: State of the art and emerging trends, ACM Transactions on Internet Technology (TOIT) 16 (2016) 1–25.
- [3] M. Lippi, P. Palka, G. Contissa, F. Lagioia, H.-W. Micklitz, G. Sartor, P. Torroni, Claudette: an automated detector of potentially unfair clauses in online terms of service, Artificial Intelligence and Law 27 (2019) 117–139.
- [4] J. K. Kummerfeld, Slate: A super-lightweight annotation tool for experts, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2019, pp. 7–12.
- [5] J. Yang, Y. Zhang, L. Li, X. Li, Yedda: A lightweight collaborative text span annotation tool, in: Proceedings of ACL 2018, System Demonstrations, 2018, pp. 31–36.
- [6] J.-C. Klie, M. Bugert, B. Boullosa, et al., The INCEpTION platform: Machine-assisted and knowledge-oriented interactive annotation, in: Proc. of the 27th International Conference on Computational Linguistics: System Demonstrations, 2018, pp. 5–9.
- [7] R. E. De Castilho, E. Mujdricza-Maydt, S. M. Yimam, et al., A web-based tool for the integrated annotation of semantic and syntactic structures, in: Proc. of the Workshop on LT4DH, 2016, pp. 76–84.
- [8] M. Wolf, D. Rüter, A. G. D’Sa, L. Reiners, J. Alexandersson, D. Klakow, Human: Hierarchical universal modular annotator, in: EMNLP, 2020.
- [9] H. Feild, T. Amello, P. Lombardo, Entities: An interface for annotating ties between entities in text, in: Proceedings of the 2020 Conference on Human Information Interaction and Retrieval, 2020, pp. 442–446.
- [10] A. Loreggia, S. Mosco, A. Zerbinati, Sentag: a web-based tool for semantic annotation of textual documents, in: To appear in the Proceedings of the 36th AAAI Conference on Artificial Intelligence 2022, AAAI Press, 2022.
- [11] A. Leff, J. T. Rayfield, Web-application development using the model/view/controller design pattern, Proceedings Fifth IEEE International Enterprise Distributed Object Computing Conference (2001) 118–127.
- [12] K. Krippendorff, Agreement and information in the reliability of coding, Communication Methods and Measures 5 (2011) 93–112.