

WARP-Text: A Web-Based Tool for Annotating Relationships Between Pairs of Texts

Venelin Kovatchev^{1,3}, M. Antònia Martí^{1,3}, Maria Salamó^{2,3}

¹Facultat de Filologia, Universitat de Barcelona

²Facultat de Matemàtiques i Informàtica, Universitat de Barcelona

³Universitat de Barcelona Institute of Complex Systems

Gran Vía de les Corts Catalanes, 585, 08007 Barcelona, Spain

{vkovatchev, amarti, maria.salamo}@ub.edu

Abstract

We present WARP-Text, an open-source web-based tool for annotating relationships between pairs of texts. WARP-Text supports multi-layer annotation and custom definitions of inter-textual and intra-textual relationships. Annotation can be performed at different granularity levels (such as sentences, phrases, or tokens). WARP-Text has an intuitive user-friendly interface both for project managers and annotators. WARP-Text fills a gap in the currently available NLP toolbox, as open-source alternatives for annotation of pairs of text are not readily available. WARP-Text has already been used in several annotation tasks and can be of interest to the researchers working in the areas of Paraphrasing, Entailment, Simplification, and Summarization, among others.

1 Introduction

Multiple research fields in NLP have pairs of texts as their object of study: Paraphrasing, Textual Entailment, Text Summarization, Text Simplification, Question Answering, and Machine Translation, among others. All these fields benefit from high quality corpora, annotated at different granularity levels. However, existing annotation tools have limited capabilities to process and annotate such corpora. The most popular state-of-the-art open source tools do not natively support pairwise annotation and require significant adaptations and modifications of the code for such tasks.

We present the first version of WARP-Text, an open source¹ web-based annotation tool, created and designed specifically for the annotation of relationships between pairs of texts at multiple layers and at different granularity levels. Our objective was to create a tool that is functional, flexible, intuitive, and easy to use. WARP-Text was built using PHP and MySQL standard implementation.

WARP-Text is highly configurable: the administrator interface manages the number, order, and content of the different annotation layers. The pre-built layers allow for custom definitions of labels and granularity levels. The system architecture is flexible and modular, which allows for the modification of the existing layers and the addition of new ones.

The annotator interface is intuitive and easy to use. It does not require previous knowledge or extensive annotator training. The interface has already been used in the task of annotating atomic paraphrases (Kovatchev et al., 2018) and is currently being used on two annotation tasks in Text Summarization. The learning process of the annotators was quick and the feedback was overwhelmingly positive.

The rest of this article is organized as follows. Section 2 presents the Related Work. Section 3 describes the architecture of the interface, the annotation scheme, the usage cases, and the two interfaces: administrator and annotator. Finally, Section 4 presents the conclusions and the future work.

2 Related Work

In the last several years, the NLP community has shown growing interest in tools that are web-based, open source, and multi-purpose: WebAnno (Yimam and Gurevych, 2013), Inforex (Marcinićzuk et al., 2017), and Anafora (Chen and Styler, 2013). Other popular non web-based annotation systems include

This work is licensed under a Creative Commons Attribution 4.0 International License. License details: <http://creativecommons.org/licenses/by/4.0/>

¹The code is available at <https://github.com/venelink/WARP> under Creative Commons Attribution 4.0 International License.

GATE (Cunningham et al., 2011) and AnCorPipe (Bertrán et al., 2008). These systems are intended to be feature-rich and multi-purpose. However, in many tasks, it is often preferable to create a specialized annotation tool to address problems that are non-trivial to solve using the multi-purpose annotation tools. One such problem is working with multiple texts in parallel. While multi-purpose annotation tools can be adapted for such use, this often leads to a more complex annotation scheme, complicates the annotation process, requires additional annotator training and post-processing of the annotated corpora. Toledo et al. (2014) and more recently Vivi Nastase and Frank (2018), Vuk Batanovi and Nikoli (2018), and Arase and Tsujii (2018) emphasize the lack of a feature-rich open-source tool for annotation of pairs of texts². Some of these authors develop simple custom-made tools with limited re-usability, designed for carrying out one specific annotation task. WARP-Text aims to address this gap in the NLP toolbox by providing a feature rich system which could be used in all these annotation scenarios.

To the best of our knowledge, the only existing multi-purpose tool that is designed to work with pairs of text and allows for detailed annotation is CoCo (España Bonet et al., 2009). It has already been used for annotations in paraphrasing (Vila et al., 2015) and plagiarism detection (Barrón-Cedeño et al., 2013). However, CoCo is not open source and is currently not being supported or updated.

3 WARP-Text

By addressing various limitations of existing tools, WARP-Text fills a gap in the state-of-the-art NLP toolbox. It offers project managers and annotators a rich set of functionalities and features: the ability to work with pairs of texts simultaneously; multi-layer annotation; annotation at different granularity levels; annotation of discontinuous scope and long-distance dependencies; and the custom definition of relationships. WARP-Text consists of two separate web interfaces: annotator and administrator. In the *administrator interface* the project manager configures the annotation scheme, defines the relationships and sets all parameters for the annotation process. The annotators work in the *annotator interface*.

WARP-Text is a tool for qualitative document annotation. It provides a wide range of configuration options and can be used for fine-grained annotation. It is best suited to medium sized corpora (containing thousands of small documents) and is not fully optimized for processing, analyzing, searching, and annotating large corpora (containing millions of documents). WARP-Text has full UTF-8 support and is language independent in the sense that it can handle documents in any UTF-8 supported natural language. So far it has been used to annotate texts in English, Bulgarian (Cyrillic), and Arabic.

WARP-Text is a multi-user system and provides two different forms of interaction between the different annotators. In the *collaborative mode*, multiple annotators work on the same text and each annotator can see and modify the annotations of the others. In the *independent mode*, the annotators perform the annotation independently from one another. The different annotations can then be compared in order to calculate inter-annotator agreement.

3.1 Annotation Scheme

The atomic units of the annotation scheme in WARP-Text are *relationships*. The properties of the *relationships* are *label* and *scope*. The *scope* of a *relationship* is a list of continuous or discontinuous *elements* in each of the two texts. The granularity level of the scope determines the *element* type. An *element* can be the whole text, a sentence, a phrase, a token, or can be defined manually. A *layer* in WARP-Text is a set of relationships, whose scopes belong to the same granularity level³. The definition of relationships and their grouping into layers is fully configurable through the administrator interface. WARP-Text supports multi-layer annotation. That is, the same pair of texts can be annotated multiple times, at different granularity levels and using different sets of relationships.

²See also the discussion about looking for tools for annotating pairs of texts in the Corpora Mailing List (May 2017): <http://mailman.uib.no/public/corpora/2017-May/026526.html> - <http://mailman.uib.no/public/corpora/2017-May/026619.html>

³There is no one-to-one correspondence between granularity level and annotation layer. Each annotation layer is a sub-task in the main annotation task. Multiple annotation layers can work at the same granularity level. For example: at layer (1) the annotator annotates the semantic relations between the tokens in the two texts; at layer (2) the annotator annotates the scope of negation and the negation cues in the two texts. Both layer (1) and layer (2) work at the token granularity level.

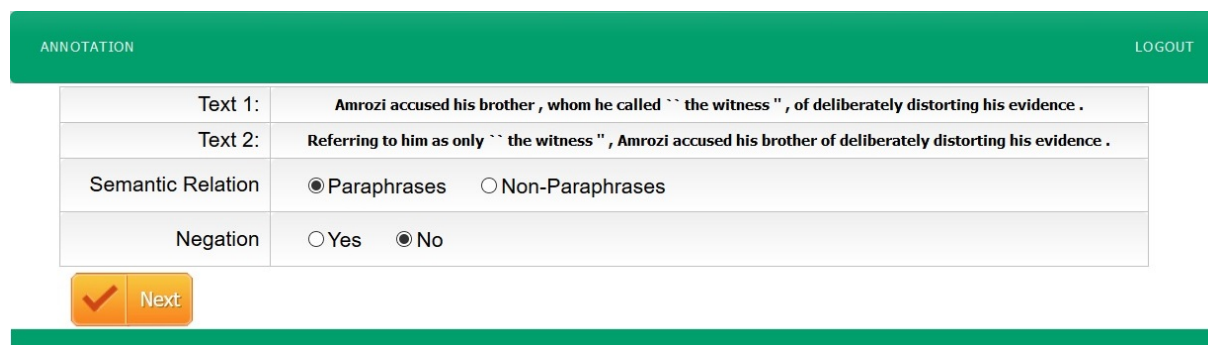
3.2 Administrator Interface

The administrator interface has three main modules: a) the *dataset management module*, b) the *user management module*, and c) the *layer management module*. In the *dataset management module* the project manager can: a) import a corpus, in a delimited text format, for annotation; b) monitor the current annotation status and statistics; and c) export the annotated corpus as an SQL file or an XML file. In the *user management module* the project manager creates new users and modifies existing ones. In this module the project manager also distributes the tasks (pairs) among the annotators. In the *layer management module* the project manager configures each of the layers and determines the order of the layers in the annotation process. The project manager configures for each individual layer: 1) the granularity level; 2) the relationships that belong to the layer; 3) the sub-relationships or properties of the relationships; 4) optional parameters such as “sentence lock” and “display previous layers”.

3.3 Annotator Interface

The annotator interface has three main modules: a) the *annotation statistics module*, b) the *review annotations module*, and c) the *annotation panel module*. In the *annotation statistics module* the annotator monitors the progress of the annotation and sees statistics such as the number of annotated pairs, and the remaining number of pairs. In the *review annotations module* the annotator reviews the text pairs (s)he already annotated and introduces corrections where necessary. The *annotation panel module* is the core of the annotator interface. One of our main objectives in the creation of WARP-Text was to make it easier to use for the annotators and to optimize the annotation time. For that reason we have made the *annotator panel module* as automated as possible and have limited the intervention of annotators to a minimum. The *annotation panel module* is generated dynamically, based on the user and project configuration. It loads the first text pair, assigned to the current annotator and guides the annotator through the different layers in the order specified by the project manager. Once the text pair has been annotated at all configured layers, the module updates the database, loads the next pair and repeats the process.

We illustrate the annotation process with the interface configuration that was used in the annotation of the Extended Typology Paraphrase Corpus (ETPC) (Kovatchev et al., 2018). The annotation scheme of ETPC consists of two layers: one layer that is configured for annotation at the text granularity level; and one layer that is configured for annotation at the token granularity level.



The screenshot shows a web interface for text annotation. At the top, there is a green header bar with the word "ANNOTATION" on the left and "LOGOUT" on the right. Below the header is a table with two columns. The first column contains labels for text and relationship types, and the second column contains the corresponding text and options. Below the table is a button with a checkmark icon and the word "Next".

Text 1:	Amrozi accused his brother , whom he called `` the witness '' , of deliberately distorting his evidence .
Text 2:	Referring to him as only `` the witness '' , Amrozi accused his brother of deliberately distorting his evidence .
Semantic Relation	<input checked="" type="radio"/> Paraphrases <input type="radio"/> Non-Paraphrases
Negation	<input type="radio"/> Yes <input checked="" type="radio"/> No

Figure 1: Annotating relationships at textual level.

The textual layer (Figure 1) displays the two texts and allows the annotator to select the values for an arbitrary number of relationships between the texts. In the case of ETPC, the two textual relationships that we were interested in were: 1) “The semantic relationship between the two texts”: “Paraphrases” or “Non-paraphrases”; and 2) “The presence of negation in either of the two sentences”: “Yes” or “No”. In ETPC, both relationships had two possible options, however WARP-Text supports multiple options for each relationship. In this first layer, the scope of the relationship is the whole text.

The second layer (Figure 2) has five functional parts, labeled in the figure with numbers from 1 to 5. The annotator can see the two texts in (1), the annotation at the previous layers in (2), and at the annotation at the current layer in (4). (3) is the navigation panel between the different layers. Finally, (5)

ANNOTATION		LOGOUT	
1	Text 1:	Amrozi accused his brother , whom he called `` the witness '' , of deliberately distorting his evidence .	
	Text 2:	Referring to him as only `` the witness '' , Amrozi accused his brother of deliberately distorting his evidence .	
2	Semantic Relation	Paraphrases	
	Negation	No	
3	<input checked="" type="button" value="Previous"/> <input checked="" type="button" value="Next"/>		
CURRENT ANNOTATION			
Type	Scope	Key	Actions
L_SameP_Sub_C			<input type="button" value="DELETE"/>
	Text 1	whom	n/a
	Text 2	to him	n/a
ADD TYPE			
5	<input type="button" value="Morphology"/>	<input type="button" value="Inflectional Changes"/>	<input checked="" type="button" value="Add Type"/>

Figure 2: Annotating relationships at token level.

is where the annotator can choose to add a new relationship. The list of possible relationships is defined by the project manager in the administrator interface. In the case of ETPC we organized the relationships in a two-level hierarchical system based on their linguistic meta-category. The token-layer annotation is more complex than the textual-layer annotation as it requires the annotation of scope in addition to the annotation a label⁴. When the annotator chooses a relationship, the "Add Type" button goes to the scope selection page (Figure 3). The scope can be discontinuous and can include elements from one of the texts only or from both. In the case of ETPC, the elements that the annotator can select are tokens. In other configurations, they can be phrases or sentences.

ANNOTATION		LOGOUT
MARK ALL THE ELEMENTS THAT BELONG TO RELATION TYPE SAME POLARITY SUBSTITUTION (CONTEXTUAL)		
Text 1:	Amrozi accused his brother , whom he called `` the witness '' , of deliberately distorting his evidence .	Whole text
Text 2:	Referring to him as only `` the witness '' , Amrozi accused his brother of deliberately distorting his evidence .	Whole text
<input checked="" type="button" value="Add Type"/>		

Figure 3: Scope selection page.

The flexibility of WARP-Text makes it easy to adapt for multiple tasks. The textual layer can be used in tasks such as the annotation of textual paraphrases, textual entailment, or semantic similarity. The atomic level annotation layer has even more applications. As we showed in ETPC, it can be used to annotate fine-grained similarities and differences between pairs of texts. It can also be used for tasks such as manual correction of text alignment. Another possible use is, given a summary or a simplified text, to identify in the reference text the exact sentences or phrases which are summarized or simplified.

4 Conclusions and Future Work

In this paper we presented WARP-Text, a web-based tool for annotating relationships between pairs of texts. Our software fills an important gap as the high quality annotation of pairwise corpora at different

⁴The token level annotation layer is an instance of the more general "atomic level annotation layer". The organization and work flow described here are the same when the granularity level is "paragraph", "sentence", "phrase", or custom defined.

granularity levels is needed and can benefit multiple fields in NLP. Previously available tools are not well suited for the task, require substantial modification, or are hard to configure. The main advantages of WARP-Text are that it is feature-rich, open source, highly configurable, and intuitive and easy to use.

As future work, we plan to add several functionalities to both interfaces. In the administrator interface, we plan to offer project managers tools for visualization and data analysis, and automatic calculation of inter-annotator agreement. In the annotator interface, we plan to fully explore the advantages of multi-layer architecture. By design, WARP-Text can support parent-child dependencies between layers. However, the pre-built modules available in this first release of the tool use only independent layers. That is, the annotation at one layer does not affect the configuration of the other layers. We also plan to explore the possibility of incorporating external automated pre-processing tools.

Acknowledgements

We would like to thank dr. Irina Temnikova and Ahmed AbuRa'ed for their support and suggestions, and the anonymous reviewers for their feedback and comments.

This work has been funded by Spanish Ministry of Economy Project TIN2015-71147-C2-2, by the CLiC research group (2017 SGR 341), and by the APIF grant of the first author.

References

- Yuki Arase and Jun'ichi Tsujii. 2018. Spade: Evaluation dataset for monolingual phrase alignment. In *Proceedings of LREC-2018*.
- Alberto Barrón-Cedeño, Marta Vila, M. Antònia Martí, and Paolo Rosso. 2013. Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics*, 39(4):917–947.
- Manuel Bertrán, Oriol Borrega, Marta Recasens, and Bàrbara Soriano. 2008. Ancorapipe: A tool for multilevel annotation. *Procesamiento del Lenguaje Natural*, 41.
- Wei-Te Chen and Will Styler. 2013. Anafora: A web-based general purpose annotation tool. In *HLT-NAACL*.
- Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damjanovic, Thomas Heitz, Mark A. Greenwood, Horacio Sagion, Johann Petrak, Yaoyong Li, and Wim Peters. 2011. *Text Processing with GATE (Version 6)*.
- Cristina España Bonet, Marta Vila Rigat, Horacio Rodríguez, and Antonia Martí. 2009. Coco, a web interface for corpora compilation.
- Venelin Kovatchev, M. Antònia Martí, and Maria Salamó. 2018. Etpc - a paraphrase identification corpus annotated with extended paraphrase typology and negation. In *Proceedings of LREC-2018*.
- Michał Marcińczuk, Marcin Oleksy, and Jan Kocóń. 2017. Inforex - a collaborative system for text corpora annotation and analysis. In *Proceedings of RANLP-2017*, September.
- Assaf Toledo, Stavroula Alexandropoupou, Sophie Chesney, Sophia Katrenko, Heidi Klockmann, Pepijn Kokke, Benno Kruit, and Yoad Winter. 2014. Towards a semantic model for textual entailment. In Cleo Condoravdi, Valeria de Paiva, and Annie Zaenen, editors, *Linguistic Issues in Language Technology vol. 9*.
- Marta Vila, Manuel Bertran, M. Antònia Martí, and Horacio Rodríguez. 2015. Corpus annotation with paraphrase types: new annotation scheme and inter-annotator agreement measures. *Language Resources and Evaluation*.
- Devon Fritz Vivi Nastase and Anette Frank. 2018. Demodify: A dataset for analyzing contextual constraints on modifier deletion. In *Proceedings of LREC-2018*.
- Pavel Vondika. 2014. Aligning parallel texts with intertext. In *Proceedings of LREC-2014*, Reykjavik, Iceland, may.
- Milo Cvetanovi Vuk Batanovi and Boko Nikoli. 2018. Fine-grained semantic textual similarity for serbian. In *Proceedings of LREC-2018*.
- Seid Muhie Yimam and Iryna Gurevych. 2013. Webanno: A flexible, web-based and visually supported system for distributed annotations. In *In Proceedings of ACL-2013 System Demonstrations*, pages 1–6.