

or properties of the relationships; 4) optional parameters such as “sentence lock” and “display previous layers”.

4.3.3 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., 2018a]. 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.

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

Next

Figure 4.1: Annotating relationships at textual level.

The textual layer (Figure 4.1) 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.

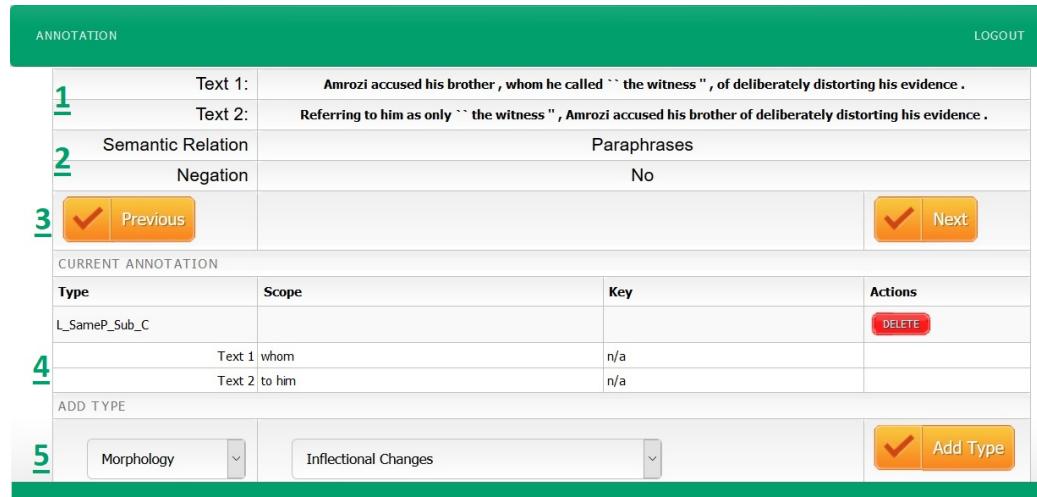


Figure 4.2: Annotating relationships at token level.

The second layer (Figure 4.2) 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) 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 4.3). 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.

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,

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.

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.	<input type="checkbox"/> Whole text
Text 2:	Referring to him as only "the witness", Amrozi accused his brother of deliberately distorting his evidence.	<input type="checkbox"/> Whole text
<input checked="" type="button"/> Add Type		

Figure 4.3: Scope selection page.

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.

4.4 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 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.