Unleashing annotations with TextAnnotator: Multimedia, multi-perspective document views for ubiquitous annotation

Giuseppe Abrami[*1], **Alexander Henlein**[*2], **Andy Lcking**[*3], **Attila Kett**[*4], **Pascal Adeberg**[*5], **and Alexander Mehler**[*6]

[*]Text Technology Lab, Goethe-University Frankfurt
[1]abrami@em.uni-frankfurt.de
[2]henlein@em.uni-frankfurt.de
[3]luecking@em.uni-frankfurt.de
[4]s7884917@stud.uni-frankfurt.de
[5]cay.ad@hotmail.de
[6]mehler@em.uni-frankfurt.de

Abstract

We argue that mainly due to technical innovation in the landscape of annotation tools, a conceptual change in annotation models and processes is also on the horizon. It is diagnosed that these changes are bound up with multi-media and multi-perspective facilities of annotation tools, in particular when considering virtual reality (VR) and augmented reality (AR) applications, their potential ubiquitous use, and the exploitation of externally trained natural language pre-processing methods. Such developments potentially lead to a dynamic and exploratory heuristic construction of the annotation process. With TEXTANNOTATOR an annotation suite is introduced which focuses on multi-mediality and multi-perspectivity with an interoperable set of task-specific annotation modules (e.g., for word classification, rhetorical structures, dependency trees, semantic roles, and more) and their linkage to VR and mobile implementations. The basic architecture and usage of TEXTANNOTATOR is described and related to the above mentioned shifts in the field.

1 Motivation

Annotation in and for computational linguistics (Gries and Berez, 2017) underwent technical and conceptual developments from XML-based annotation formats to integrated GATE (Cunningham et al., 2013) or UIMA (Götz and Suhre, 2004) frameworks (Wilcock, 2017). One reason for that development is that annotation (regardless of the annotated media such as texts, images, music, video, and so on) is bound to annotation tools, usually one annotation tool per annotation task or purpose (Cassidy and Schmidt, 2017; Dipper et al., 2004). Annotation tools are themselves subject to (technical)

development (see, for instance, the annotation of rhetorical relations (Helfrich et al., 2018)). Furthermore, annotation is often part of a machine learning (ML) pipeline where machine learned applications are trained on annotated data (Rumshisky and Stubbs, 2017), so that they can later perform annotations automatically on larger data sets. This is most explicitly expressed in the MATTER/MAMA annotation model (Pustejovsky and Stubbs, 2012). In order to secure interoperability and data exchange in this dynamic landscape, annotations of linguistic phenomena (should) follow a standard (e.g. ISO, 2016).

As has been observed by Finlayson and Erjavec (2017), there are still features that are missing or only seldomly addressed in annotation tools. So further developments are here to be expected.

However, we argue that another technical and conceptual change takes place, a change that is characterised by the following, partly mutually influencing, features.

Multi-Mediality and -Perspectivity. An annotation tool trivially is a medium (for annotation). Now, as is known, for instance, from readability research, the "physical" properties of the medium *text* influence text processing: no readability difference between serif and sans serif font types has been observed (Ali et al., 2013), but they seem to differ with respect to information recall (Gasser et al., 2005). Likewise, the choice of document preparation system has an effect on the efficiency and satisfaction of the document preparer (Knauff and Nejasmic, 2015). Transferred to annotation tools, such findings evince that users may produce different results with different annotation media. Taking advantage of this effect, annotation tools

should offer multiple views on the same data: an attribute called *Multi-Perspectivity* and realized by the tool's *Multi-Mediality*. Multi-Mediality and -Perspectivity can be realized in various ways, ranging from low-level customizable display properties to high-level exploratory means of inspecting a certain kind of data with tools/views that are developed for different data types. We conjecture both heuristic and error-reducing gains by multi-media, multi-perspective methods.

Note that multi-perspectivity is different from multimodal annotation as carried out by using video (hence the attribute 'multimodal') annotation tools such as ANVIL (Kipp, 2014) or ELAN (Wittenburg et al., 2006): while multimodality tools allow the analysis of visually recorded communication settings, multi-perspectivity tools render the same input data in various formats.

VR and AR annotations. Multi-Mediality comprises virtual reality (VR) and augmented reality (AR) as special cases. So the claims made in the previous paragraph apply here, too. However, annotating in VR or AR has some obvious repercussions on human-computer interaction (HCI, where "computer" stands for the annotation tool used). Most notably, classic HCI interfaces such as a computer mouse are replaced by locomotion or (virtual) manipulation. Again, a heuristic effect is to be conjectured, but such "immersive annotation settings" have still to be explored. A consequence is already visible, however, namely that the range of annotation objects is extended: real-world objects (AR) and the annotators' actions (VR) become potential subjects of annotations. The former is, for instance, needed in geospatial information systems (cf. Sec. 2); the latter can be used to label professional actions as learned, for instance, in virtual nurse education (Plotzky et al., 2021). We also note that VR systems are still quite new in the computational linguistics community. However, as such systems spread to all areas of human communication, people will become accustomed to their use, and the current gap between the use of traditional systems and VR will naturally disappear.

Ubiquity. Porting annotation software to mobile phones cuts any locational constraints on annotators (given a sufficient internet infrastructure). Mobile annotation probably unfold their potential when embedded into *games with a purpose* (von Ahn, 2006): annotators produce annotations "for fun" and *en passant*, when, say, being on a travel. Mobile annotations combines with AR annotations, leading to a *qualitative* (not just quantitative) change in the units of annotation.

ML for annotation (or: human-in-the-loop). The predominant annotation model conceives annotation as a means for providing data for machine learning. And annotations will surely continue to be produced and used in this way. However, the current computational annotation landscape also treads the opposing path: pre-trained ML tools are used for automatic (large-scale) annotation of documents which are then corrected by human annotators (de Castilho et al., 2019; Hemati et al., 2016). Accordingly, the role of human annotators changes from "mere" data-generators (Consten and Loll, 2012) or "two-legged meters" (Cohen, 1960) to "humans-in-the-loop" (Wagner, 2016) (i.e., a post-editing phase is interspersed at some point into the ML process, a.k.a *active learning*, Cohn et al., 1994; Settles, 2012).

Dynamics of annotation processes. It is well-known that due to an interplay of theoretical knowledge and data structure of annotation units, (linguistic) annotations exhibit a "circular" trait (Consten and Loll, 2012) – this is also reflected in the iterative design of the MAMA cycle (Pustejovsky and Stubbs, 2012). Annotation manuals and especially standardizations like the Semantic Annotation Framework (SemAF) (ISO, 2016) are means for taming this process. In fact, however, in particular the Multi-Mediality and -Perspectivity fosters the circularity of annotation processes since viewing one document from different viewpoints is a heuristic activity (cf. "Multi-Modality and -Perspectivity"). There are two consequences of this situation: Firstly, the dynamic nature of annotations is emphasized. This includes to construe annotations as parts of sequences of annotations instead of as singular tasks (cf. the argument from circularity, triggered by the mutual theoretical pre-conception and the actual structure of annotation data) – regardless of whether the encompassing sequence tasks are actually carried out. In other words: in designing an annotation task both (implicitly) presupposed and (potential) follow-up annotations have to be kept in mind. This is already partly reflected, for instance, in the plug-ins approach to dialogue act annotation (Bunt, 2019).

Secondly, even dynamic annotation processes

cannot afford to ignore achieved standards. On the one hand, multi-media and multi-perspective annotation tools support established schemes and ontologies. On the other hand, best practices and process standards will emerge from dynamic annotation processes.

Interim conclusion We anticipate a potential shift in thinking of and carrying out annotations, as indicated in the "Dynamics of annotation processes. This shift is driven by technological achievements mainly in the domain of VR/AR, extended pre-processing, and ubiquitous computing." Preliminary (i.e., as long as a corresponding full-blown annotation model has been developed) we refer to a system that exhibits the envisaged facilities as MUVAMP (**M**ulti-Mediality and -Perspectivity, **U**biquity, **VR/AR**, **ML**, **P**rocess-orientedness). Given that this is a preliminary characterization, it is obvious that no current annotation system fulfils MUVAMP. However, to making the envisaged shift happen, a precondition seems to be an annotation tool that hosts several modules (otherwise it remains unclear how multi-perspectivity is achieved). In the following we introduce TEXTANNOTATOR as a MUVAMP-oriented annotation suite for unleashing annotations along the above lines. After reviewing related approaches, we present TEXTANNOTATOR module-wise and indicate each module's role for MUVAMP.

2 Related Work

There are applications around that address some of the features outlined above. We are aware of the following ones:

- Incorporating machine learning applications into the annotation pipeline is carried out in INCEpTION (de Castilho et al., 2019) (which extends on *WebAnno* (Eckart de Castilho et al., 2016)), the commercial service *prodi.gy* (Montani and Honnibal, 2018) and the TEXTIMAGER (Hemati et al., 2016) (the latter also underlies the present work).
- Annotation in virtual reality is implemented by means of a note taking facility in (industrial) VR environments (Clergeaud and Guitton, 2017). VR visualisations have also been used in the study of multimodal referring expressions (Pfeiffer, 2012). In Wither et al. (2009), an annotation taxonomy and a prototype study on outdoor augmented reality annotation is developed.

- Mobile annotation of geospatial information is made available in MobiTOP (HoeLian Goh et al., 2012). The mobile annotation of images, e.g. for social media uses, is enabled by Anguera et al. (2008).

3 TextAnnotator

TEXTANNOTATOR is a suitable candidate as a multimedia and multimodal annotation environment for UIMA documents (Götz and Suhre, 2004). The UIMA-based annotations are driven by the TEXTANNOTATOR as a RESTfull application developed in Java. Documents which are not available in UIMA can be transferred into this format by using TEXTIMAGER (Hemati et al., 2016), which provides a rich machine learning backend for automatic annotation accouting for the 'ML' component of MUVAMP. The UIMA documents are stored through the *UIMADatabaseInterface* (Abrami and Mehler, 2018) within MongoDB[1] and can be used simultaneously and collaboratively through TEXTANNOTATOR. Collaborativity and simultaneity are enabled with bidirectional information exchange via web-socket between TEXTANNOTATOR and all client systems and is an important component for ubiquitous use. In addition, the web-socket allows other annotation tools to be connected to TEXTANNOTATOR, to ensure its multimedia nature (see Sec. 4 and 5). The connection between TEXTANNOTATOR and its client systems is illustrated in Fig. 2. Annotations stored in UIMA documents are organized in different *annotation views* (AV). Each of these views contains different annotations and is related to a specific topic or user. For each annotator, a user view is created when a document is initially opened, which duplicates the original annotations. Thus, each AV shows a different perspective, state, or context on the same document. Furthermore, the different AVs allow the computation of inter-annotator agreement, which enables to assess the consistency of annotations in a project (Krippendorff, 2018), based on user permissions.

In addition, all annotations can be used independently of TEXTANNOTATOR: they can be completely downloaded for further processing. The reuse of the annotations as a basis for ML is thus customizable, depending on the needs of the particular application. In the following, we show how TEXTANNOTATOR accounts for **Multi-Mediality and -Perspectivity**, **Dynamics of annotation pro-**

---

[1] https://www.mongodb.com/

cesses, and **ML for annotation**. **VR and AR**, and **Ubiquitous use** are dealt with in Sec. 4 and 5, respectively.

QuickAnnotator still allows rapid annotation of named entities and words and multi-token expression in general through a simple selection of a target class and subsequent assignment when clicking on tokens (Abrami et al., 2019). To increase annotation performance, a recommendation function was implemented that allows the selected target class, based on the token's lemma, to be applied to all other tokens of the same lemma in the same document, paragraph, or current sentence. In addition, all tokens annotated by this function are marked so that annotators can easily target and post-process them.

Another function is the combination of tokens to multitokens: By now this function has been extended with the possibility to separate tokens at any position as well as the capability to correct OCR errors (see Fig. 1). This user-friendly function, which can also be executed via drag & drop, enables the correction of incorrectly recognized token boundaries which is a frequent and popular error, especially in the context of OCR recognition of texts.

As it provides basic corrections to the texts and establishes the prerequisites for future annotation processes, QUICKANNOTATOR develops into a pre-editing tool which is employed before the main annotation work is done with more specific tools such as PROPANNOTATOR or DEPANNOTATOR. Hence, the triplet of QUICKANNOTATOR, PROPANNOTATOR, and DEPANNOTATOR gives rise to multi-
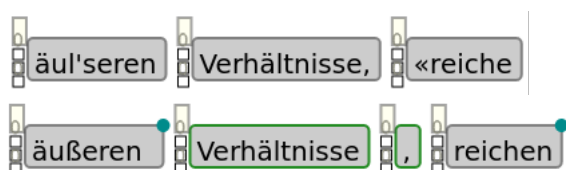

Figure 1: Tokens can be corrected as required. Firstly, incorrect token boundaries can be split using a simple key combination. In the present example this was done with the token merged with the comma (green border). Secondly, OCR errors can be corrected in QUICKANNOTATOR by clicking on the corresponding tokens. In this way the original text is not changed, but a correction token is generated, which is placed on top of the affected tokens. A corrected token is visualized with a green dot in the upper right corner. By moving the cursor over it, the original token is shown.

perspectivity (in particular if annotation files are preprocessed in such a way that they contain annotation layers according to the corresponding tools' specification).

PropAnnotator uses relations adopted from the SemAF standard (ISO, 2014a) as well as from Prop-Bank (Palmer et al., 2005) (the latter can be mapped onto the former (Ide et al., 2017, 133)). To this end, we converted the structures defined in these standards into a UIMA type system[2]. Since the last presentation of PROPANNOTATOR in Abrami et al. (2019), significant improvements and new features have been implemented. The underlying data model allows for annotating a wide range of relations a subset of which is available in the current web interface of PROPANNOTATOR:

- Argument and modifier relations (following PropBank);
- Time relations (temporal entities and TLinks from ISOTimeML (Pustejovsky, 2017; ISO, 2012));
- Spatial-Relations (Qualitative Spatial Links (QSLinks) from ISOSpace (Pustejovsky et al., 2011; ISO, 2020a));
- A few custom extensions (for example, labeling idiomatic expressions and separated verb particles in German).

These relations are used in order to carry out semantic role labeling. Regarding semantic role labeling, the annotation of functional roles (argument and modifier relations) depends on the *sense* of the verb heading the corresponding sentence (Levin, 1993). In support of this view, PROPANNOTATOR complements semantic role annotation with verb sense annotation. Verb senses are distinguished according to PropBank's frameset lexicon[3], GermaNet (Hamp and Feldweg, 1997; Henrich and Hinrichs, 2010) as well as – for evaluation purposes – E-Valbu (Schumacher et al., 2004). The cross-language mixture of sense inventories is due the fact that the main language of actual annotation documents is German, but the majority of (large-scale) resource has been developed in and for English. Hence, PROPANNOTATOR provides an annotation-based mapping between English and German verb sense. Since this mapping involves translation issues, the small but hand-crafted verb

---

[2] https://github.com/texttechnologylab/UIMATypeSystem
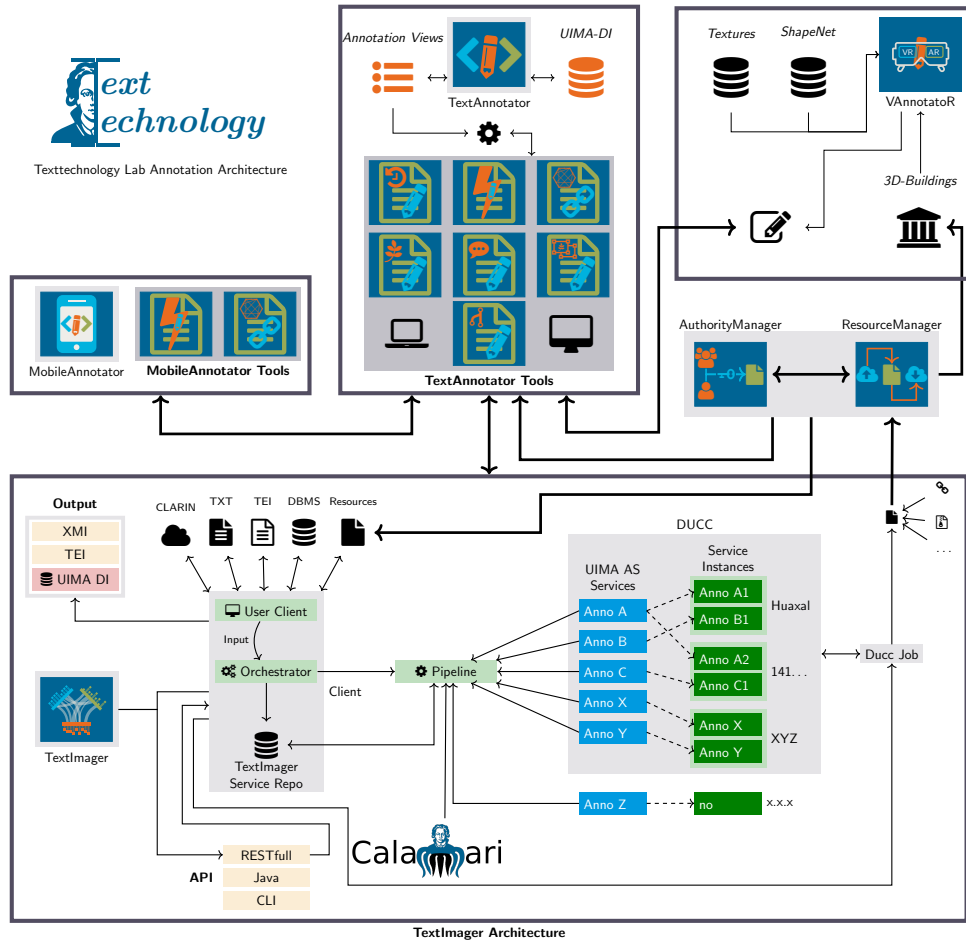[3] http://verbs.colorado.edu/propbank/framesets-english-aliases/

Figure 2: Illustration of the interrelationships and communication routes between the individual infrastructures. In the lower area the domain of TEXTIMAGER (Hemati et al., 2016) is shown, a multi-server system for automatic pre-processing of textual data based on UIMA. Several pipelines, each for different analyses, can be used to process the texts for use by TEXTANNOTATOR. Texts must exist in UIMA format in order to use these within the infrastructures. The upper architectures show the individual tools which are also considered in this paper in more detail. This simplified presentation shows the relationship between the tools, which all use TEXTANNOTATOR as a core service. The different annotation environments, MOBILEANNOTATOR, TEXTANNOTATOR and VANNOTATOR, are located in the upper area. Being the center of all manual annotation processes, TEXTANNOTATOR enables the use of TEXTIMAGER and thus to automatize parts of the annotation process. Each tool is directly or indirectly connected to the *ResourceManager* and *AuthorityManager* (Gleim et al., 2012) in order to manage the annotation of documents. All documents managed in ResourceManager are database objects manageable by the *UIMA database interface*. This usage takes place entirely within TEXTANNOTATOR. All tools that want to perform or use UIMA-based annotations are connected to TEXTANNOTATOR in order to subsequently use all implemented functions. Calamari, shown in the bottom region, is a Blazegraph (https://blazegraph.com/) implementation (still under development) for maintaining various ontologies within the TEXTANNOTATOR/TEXTIMAGER infrastructure.

sense inventory of E-Valbu is built-in as a ground truth standard of comparison.

DepAnnotator is the newest tool designed for visualization and annotation of dependency structures in texts. Based on different dependency tag sets (derived from TIGER (Brants et al., 2004), respectively NEGRA (Skut et al., 1997), and Universal Dependencies (de Marneffe et al., 2014)) existing dependencies can be deleted and new ones

can be created. In addition, as with all tools, it is also possible to manually annotate texts without pre-annotated dependency information, which is illustrated in Fig. 4.

4 MobileAnnotator

To remove the binding of the annotation situation to desktop sessions, so to speak, to enable annotations in mobile contexts, quasi **ubiquitously** (whether in sitting, standing or walking
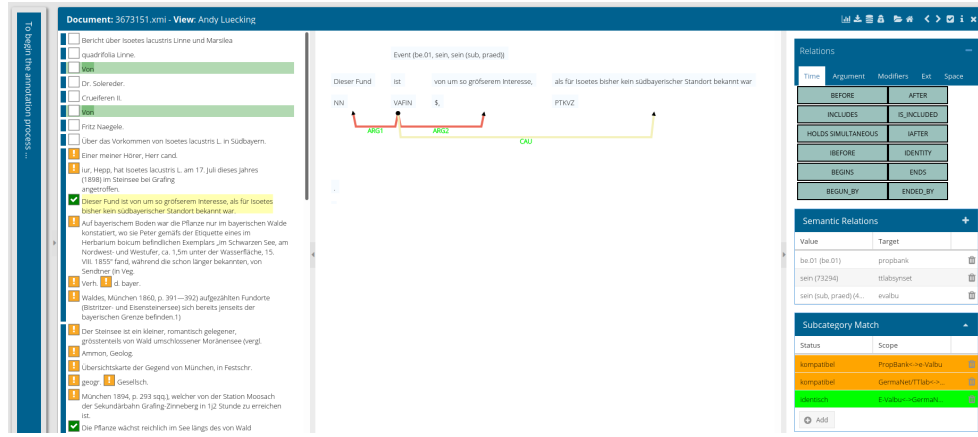
Figure 3: Visualization of PROPANNOTATOR: On the left side, the sentences of the document are displayed, which can be transferred to the middle annotation area by a click. Each sentence can be assigned with a status, which documents the annotation progress: green indicates completed sentences and yellow indicates problems, which is helpful both for later evaluation and for interrupting the process. The panel on the right side shows the annotation options. The upper part of these options shows a set of relations that can also be selected by a click. Below this is a list of annotated semantic senses, which is only enabled when an event is selected. In the center of PROPANNOTATOR's interface the annotation environment is shown by visualizing the selected sentence (left) token-wise. Similar to QUICKANNOTATOR, multitokens can be created and tokens can be separated via drag & drop. Under each token, its part of speech is displayed; clicking on a verb turns it into an "event", which can then be sense-disambiguated. Tokens can be linked semantically by drag & drop, based on the selection of a corresponding relation in the options panel (right). Colors are used to distinguish between different relation types.

position of the annotator), we have developed MO-BILEANNOTATOR (Adeberg, 2020). Based on Angular[4], we adapted two tools of TEXTANNO-TATOR (QUICKANNOTATOR and KNOWLEDGE-BASELINKER) to enable mobile access. MO-BILEANNOTATOR was developed as a TEXTAN-NOTATOR client (see Fig. 2) using its functional-ity. This allows the implementation of additional functions which are not available in the browser-based version. At the same time, documents are still accessible only after user authentication and all annotations are stored in MOBILEANNOTATOR in appropriate annotation views. The the control and use of UIMA documents is thus analogous to TEXTANNOTATOR. To motivate it with concrete examples: with MOBILEANNOTATOR, train rides, waiting time at the doctor's office, at the bus stop, or anywhere else can be used for annotation tasks. Mobile annotations, of course, attain ubiquity.

5 VAnnotatoR

VR-based annotation is provided by VANNO-TATOR, a UIMA-based annotation environment implemented in Unity3D[5]. Since VANNOTA-TOR (Spiekermann et al., 2018) is also based on

TEXTANNOTATOR (see Fig 2), its annotations can be further processed with any other annotation me-dia of TEXTANNOTATOR.

VANNOTATOR addresses a range of scenarios: visualization and interaction with historical infor-mation (Abrami et al., 2020b), annotation of texts, their interlinking with images and 3D objects, and the creation of 3D spaces enriched with hypertext functionalities (Mehler et al., 2018; Abrami et al., 2020a).

VANNOTATOR is currently extended to include SemAF-related functionalities. A pilot study of this extension is presented in (Henlein et al., 2020). The main focus is on the annotation of spatial re-lations (IsoSpace, Pustejovsky et al., 2011; ISO, 2020b), semantic roles (SrLinks, ISO, 2014a) and coreference relations (MetaLinks, ISO, 2014b). This is done to generate text-to-scene data, which in turn is used to train ML systems. Fig. 6 exem-plifies this sort of annotation data. In this example, we take advantage of the spatial capabilities of VR to automate as many spatial annotations as possible. That is, whenever the annotator arranges objects in virtual space based on their description in the underlying text, a subset of the relationships of the objects implied by this arrangement is explicitly annotated by the system itself. This concerns ob-jects that are implicitly or explicitly involved in

---

[4]https://angular.io/
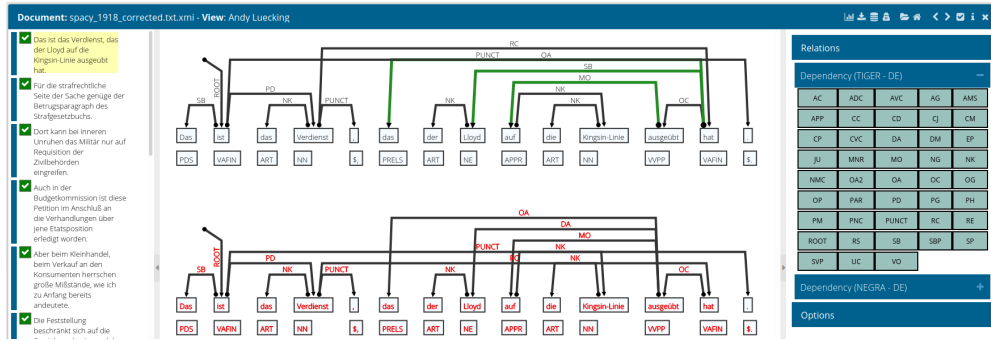[5]https://unity.com/

Figure 4: The interface of DEPANNOTATOR is similar to that of PROPANNOTATOR. On the left one finds the sentences to be annotated (together with their annotation status). The panel on the right displays selectable options subdivided into tagsets. Two visualizations of the focal sentence are displayed in the center of the window: the lower sentence shows the dependency tree created by the parser selected in TEXTIMAGER; the upper sentence shows its correction. Green lines encode selected dependency relations created by the human annotator. With DEPANNOTATOR it is possible to visually compare automatically created dependency trees and their corrections. DEPANNOTATOR additionally contains statistics for automatic comparison of such trees.

this description. In addition to placing entities in virtual space, the annotators' movements and gestures could be used in the future for this purpose. In a nutshell, VANNOTATOR meets the MUVAMP requirement for using VR and AR for the purpose of multimedia and multi-perspective annotation.

6 Application usage

The tools described so far are being used in various lectures and qualification work (e.g. Kühn (2018);

Figure 5: An excerpt from two annotation tools of MOBILEANNOTATOR. Left: adaptation of QUICK-ANNOTATOR; right: adaptation of KNOWLEDGE-BASELINKER. By selecting a token, it can be annotated. A longer activation of tokens enables the creation of multitokens.

Smaji (2020); Kett (2020); Lööck (2020)) to automatically validate annotated documents (human in the loop) or to gain new perspectives on annotated documents. In particular, TEXTANNOTATOR is used as a browser-based suite for the correction of automatic annotations generated with TEXTIMAGER.

In addition, TEXTANNOTATOR is used in the biodiversity project *BioFID* (Driller et al., 2020). This project is concerned with the semantic indexing of historical biodiversity texts. For this purpose, TEXTANNOTATOR is used to annotate texts in order to perform various linguistic analyses. Within the BioFID project, 79,813 "net" annotations[6] have been produced using QUICKANNOTATOR (Lücking et al., 2021). "Net" means the following: since within BIOfid documents are annotated by more than one annotator for the sake of assessing inter-rater agreement, one and the same annotation unit may receive a label repeatedly but from different annotators. The net count ignores such reduplications and only takes unique labels into account.

These numbers show that many annotations can be performed by different users in a very short time (for a user evaluation of one of TEXTANNOTA-TOR's modules, TREEANNOTATOR, see Helfrich et al., 2018, Sec. 3). At the same time, all annotations are available in a uniform and portable format, which ultimately simplifies external processing and reuse, e.g., for ML tasks.

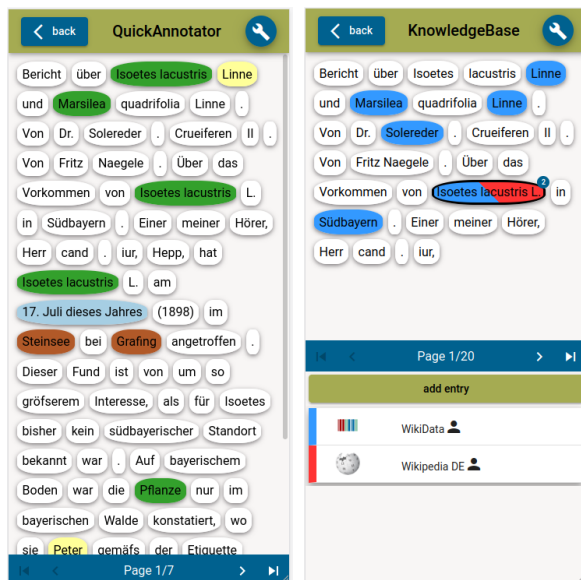The combination of a large number of different annotation functions (at the word, sentence, or text
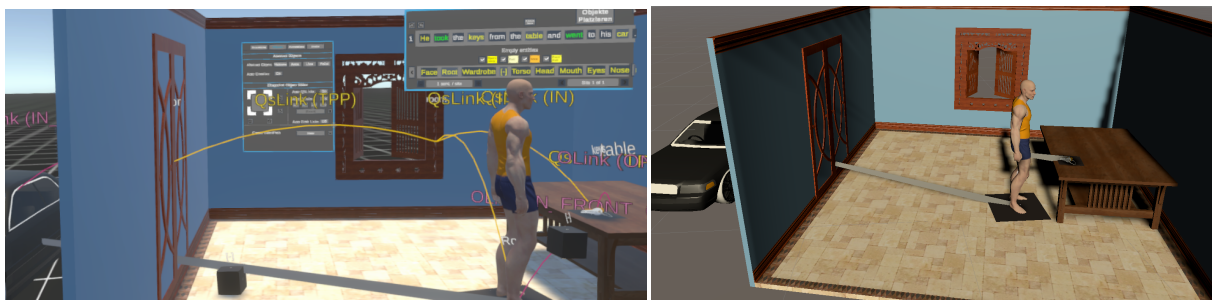
---

[6]as of 27th October 2020

Figure 6: Annotation example for the sentence: *He took the keys from the table and went to his car*. Left: VR view; Right: Rendering-View. In the VR annotation view one can see in yellow the *Qualitative Spatial Links* (QSLinks) and in red the *Orientation Links* (OLinks). The QSLinks and OLinks are mostly generated automatically. The thick gray line in both views represents the EventPath (here: key in hand, person to car).

level) that provide *multiple annotation perspectives* on the same text, as well as the *multimedia bandwidth* that comes with them, is, to our knowledge, currently unique in the field of annotation of natural language texts.

7 Future Work

Currently, not all annotation features available through TEXTANNOTATOR can be used by downstream tools (e.g. MOBILEANNOTATOR). To enable full ubiquitous use, different approaches for the different media (VR, AR, mobile devices) are required. In particular, we will consider the possibilities and limitations of AR systems and the extent to which they can be used for annotation purposes. While the available hardware is still very limited (price, availability, technical features, . . .), in the near future it will become available to the general public, similar to VR. Furthermore, in addition to the extension of VANNOTATOR's *RoomBuilder* according to the SemAF standards, an annotation environment for TEXTANNOTATOR and MOBILEANNOTATOR is planned. This extension should make it possible to pre-annotate texts at home, in the office, or on the road, to largely complete their annotation in VR – also with regard to implied annotations – and to correct and refine the results later with conventional 2D interfaces if necessary. Insofar as these annotations refer to artifacts that are visible or even traversable in reality (e.g. streets, houses, squares), this multimedia annotation process can be significantly enriched by AR functionalities, since the direct view of the objects to be annotated can compensate for inadequacies of their representation in VR.

Thanks to the large number of tools in TEXTANNOTATOR, a wide range of annotation tasks can be addressed. However, since it is inefficient in the long run to develop tools with reference to specific annotation requirements, a more dynamic approach that simplifies the planning of annotation projects suggests itself. To meet this requirement, TEXTANNOTATOR is being further developed as a tool for modeling annotation models and corresponding annotation tools. Furthermore, it is planned to publish TEXTANNOTATOR via GitHub.

8 Conclusion

We introduced the concept of MUVAMP (**M**ultimediality and -perspectivity, **U**biquity, **VR/AR**, **M**L, **P**rocess-orientation) and argued how TEXTANNOTATOR and the annotation tools around it meet this concept. Reflecting on and studying MUVAMP, and devising corresponding annotation models is still a desideratum for computational linguistics. The increasing complexity of annotation tasks and their representation in tools in order to be able to use them collaboratively and simultaneously in a homogeneous annotation environment at best. In addition, enabling annotators to use multi-perspective multimedia annotation tools is an area where established best practices do not yet exist. In order to contribute to this research perspective, we have presented the latest developments of TEXTANNOTATOR and outlined future development steps.

In conclusion, it is our strong interest to discuss and also establish with the research community a new and more innovative way in the implementation of annotation processes. For this purpose, not only concepts and procedures are necessary, but also adequate and flexible software solutions – such as TEXTANNOTATOR.

References

Giuseppe Abrami, Alexander Henlein, Attila Kett, and Alexander Mehler. 2020a. Text2SceneVR: Generating hypertexts with vannotator as a pre-processing step for text2scene systems. In *Proceedings of the 31st ACM Conference on Hypertext and Social Media*, HT '20, page 177186, New York, NY, USA. Association for Computing Machinery.

Giuseppe Abrami and Alexander Mehler. 2018. A uima database interface for managing nlp-related text annotations. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12*, LREC 2018, Miyazaki, Japan.

Giuseppe Abrami, Alexander Mehler, Andy Lcking, Elias Rieb, and Philipp Helfrich. 2019. TextAnnotator: A flexible framework for semantic annotations. In *Proceedings of the Fifteenth Joint ACL - ISO Workshop on Interoperable Semantic Annotation, (ISA-15)*, ISA-15.

Giuseppe Abrami, Alexander Mehler, Christian Spiekermann, Attila Kett, Simon Lööck, and Lukas Schwarz. 2020b. *Educational Technologies in the area of ubiquitous historical computing in virtual reality*. Taylor & Francis.

Pascal Adeberg. 2020. MobileAnnotator: an App for TextAnnotator. bachelor's thesis, Institute of Computer Science and Mathematics, Text Technology Lab, Johann Wolfgang Goethe-Universitt, Frankfurt, Germany. Original title: MobileAnnotator: eine App für den TextAnnotator.

Luis von Ahn. 2006. Games with a purpose. *Computer*, 39(6):92–94.

Ahmad Zamzuri Mohamad Ali, Rahani Wahid, Khairulanuar Samsudin, and Muhammad Zaffwan Idris. 2013. Reading on the computer screen: Does font type have effects on web text readability?. *International Education Studies*, 6(3):26–35.

Xavier Anguera, JieJun Xu, and Nuria Oliver. 2008. Multimodal photo annotation and retrieval on a mobile phone. In *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*, MIR '08, page 188194, New York, NY, USA. Association for Computing Machinery.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. TIGER: Linguistic interpretation of a German corpus. *Journal of Language and Computation*, 2:597–620.

Harry Bunt. 2019. Plug-ins for content annotation of dialogue acts. In *Proceedings of the Fifteenth Joint ACL-ISO Workshop on Interoperable Semantic Annotation*, ISA-15, pages 33–45.

Steve Cassidy and Thomas Schmidt. 2017. Tools for multimodal annotation. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 209–227. Springer Netherlands, Dordrecht.

Richard Eckart de Castilho, Éva Mújdricza-Maydt, Seid Muhie Yimam, Silvana Hartmann, Iryna Gurevych, Anette Frank, and Chris Biemann. 2016. A web-based tool for the integrated annotation of semantic and syntactic structures. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH)*, pages 76–84, Osaka, Japan. The COLING 2016 Organizing Committee.

Richard Eckart de Castilho, Nancy Ide, Jin-Dong Kim, Jan-Christoph Klie, and Keith Suderman. 2019. Towards cross-platform interoperability for machine-assisted annotation. *Genomics & Informatics*, (2).

D. Clergeaud and P. Guitton. 2017. Design of an annotation system for taking notes in virtual reality. In *2017 3DTV Conference: The True Vision – Capture, Transmission and Display of 3D Video*, 3DTV-CON, pages 1–4.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20:37–46.

David Cohn, Les Atlas, and Richard Ladner. 1994. Improving generalization with active learning. *Machine learning*, 15(2):201–221.

Manfred Consten and Annegret Loll. 2012. Circularity effects in corpus studies – why annotations sometimes go round in circles. *Language Sciences*, 34(6):702–714.

H. Cunningham, V. Tablan, A. Roberts, and K. Bontcheva. 2013. Getting more out of biomedical documents with GATE's full lifecycle open source text analytics. *PLoS Comput Biol*, 9(2).

Stefanie Dipper, Michael Götze, and Manfred Stede. 2004. Simple annotation tools for complex annotation tasks: an evaluation. In *Proceedings of the LREC Workshop on XML-based Richly Annotated Corpora*, LREC 2004, pages 54–62.

Christine Driller, Markus Koch, Giuseppe Abrami, Wahed Hemati, Andy Lücking, Alexander Mehler, Adrian Pachzelt, and Gerwin Kasperek. 2020. Fast and easy access to Central European biodiversity data with BIOfid. In *Biodiversity Information Science and Standards*, volume 4 of *BISS*.

Mark A. Finlayson and Tomaž Erjavec. 2017. Overview of annotation creation: Processes and tools. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 167–191. Springer Netherlands, Dordrecht.

Michael Gasser, Julie Boeke, Mary Haffernan, and Rowena Tan. 2005. The influence of font type on information recall. *North American Journal of Psychology*, 7(2):181–188.

Rdiger Gleim, Alexander Mehler, and Alexandra Ernst. 2012. Soa implementation of the ehumanities desktop. In *Proceedings of the Workshop on Service-oriented Architectures (SOAs) for the Humanities: Solutions and Impacts, Digital Humanities 2012, Hamburg, Germany*.

T. Götz and O. Suhre. 2004. Design and implementation of the UIMA Common Analysis System. *IBM Systems Journal*, 43(3):476–489.

Stefan Th. Gries and Andrea L. Berez. 2017. Linguistic annotation in/for corpus linguistics. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 379–409. Springer Netherlands, Dordrecht.

Birgit Hamp and Helmut Feldweg. 1997. GermaNet - a lexical-semantic net for German. In *Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*.

Philipp Helfrich, Elias Rieb, Giuseppe Abrami, Andy Lücking, and Alexander Mehler. 2018. Treeannotator: Versatile visual annotation of hierarchical text relations. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12*, LREC 2018, Miyazaki, Japan.

Wahed Hemati, Tolga Uslu, and Alexander Mehler. 2016. Textimager: a distributed uima-based system for nlp. In *Proceedings of the COLING 2016 System Demonstrations*. Federated Conference on Computer Science and Information Systems.

Alexander Henlein, Giuseppe Abrami, Attila Kett, and Alexander Mehler. 2020. Transfer of isospace into a 3d environment for annotations and applications. In *Proceedings of the 16th Joint ACL - ISO Workshop on Interoperable Semantic Annotation*, pages 32–35, Marseille. European Language Resources Association.

Verena Henrich and Erhard Hinrichs. 2010. GernEdiT - the GermaNet editing tool. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

D. HoeLian Goh, K. Razikin, C. Sian Lee, E. Peng Lim, K. Chatterjea, and C. Hung Chang. 2012. Evaluating the use of a mobile annotation system for geography education. *The Electronic Library*, 30(5):589–607.

Nancy Ide, Nicoletta Calzolari, Judith Eckle-Kohler, Dafydd Gibbon, Sebastian Hellmann, Kiyong Lee, Joakim Nivre, and Laurent Romary. 2017. Community standards for linguistically-annotated resources. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, chapter 4, pages 113–165. Springer Netherlands, Dordrecht.

ISO. 2012. Language resource management – Semantic annotation framework (SemAF) – Part 1: Time and events (SemAF-Time, ISO-TimeML). Standard ISO 24617-1:2012.

ISO. 2014a. Language resource management – Semantic annotation framework (SemAF) – Part 4: Semantic roles (SemAF-SR). Standard ISO/IEC TR 24617-4:2014.

ISO. 2014b. Language resource management Semantic annotation framework (SemAF) Part 7: Spatial information (ISO-Space). Standard ISO/IEC TR 24617-7:2014, International Organization for Standardization.

ISO. 2016. Language resource management – Semantic annotation framework (SemAF) – Part 6: Principles of semantic annotation (SemAF principles). Standard ISO 24617-6:2016(E).

ISO. 2020a. Language resource management – Semantic annotation framework – Part 7: Spatial information. Standard ISO 24617-7:2020.

ISO. 2020b. Language resource management Semantic annotation framework (SemAF) Part 7: Spatial information (ISO-Space). Standard ISO/IEC TR 24617-7:2020.

Attila Kett. 2020. text2city: Spatial visualization of textual structures. bachelor's thesis, Institute of Computer Science and Mathematics, Text Technology Lab, Johann Wolfgang Goethe-Universitt, Frankfurt, Germany. Original title: text2city: Räumliche Visualisierung textueller Strukturen.

Michael Kipp. 2014. ANVIL: A universal video research tool. In Jacques Durand, Ulrike Gut, and Gjert Kristofferson, editors, *Handbook of Corpus Phonology*, chapter 21, pages 420–436. Oxford University Press, Oxford, UK.

Markus Knauff and Jelica Nejasmic. 2015. An efficiency comparison of document preparation systems used in academic research and development. *PLOS ONE*, 10(4).

Klaus Krippendorff. 2018. *Content Analysis: An Introduction to Its Methodology*, 4 edition. SAGE Publications.

Vincent Roy Kühn. 2018. A gesture-based interface to VR. bachelor's thesis, Institute of Computer Science and Mathematics, Text Technology Lab, Johann Wolfgang Goethe-Universitt, Frankfurt, Germany.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago.

Simon Lööck. 2020. Distributed annotation in virtual reality. bachelor's thesis, Institute of Computer Science and Mathematics, Text Technology Lab, Johann Wolfgang Goethe-Universitt, Frankfurt, Germany.

Andy Lücking, Christine Driller, Giuseppe Abrami, Adrian Pachzelt, Manuel Stoeckel, and Alexander Mehler. 2021. Multiple annotation for biodiversity. Developing an annotation framework among biology, linguistics and text technology. *Language Resources and Evaluation*. Accepted with minor revisions.

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre, and Christopher D. Manning. 2014. Universal Stanford dependencies: A cross-linguistic typology. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, LREC'14, pages 4585–4592, Reykjavik, Iceland. European Language Resources Association (ELRA).

Alexander Mehler, Giuseppe Abrami, Christian Spiekermann, and Matthias Jostock. 2018. VAnnotatoR: A framework for generating multimodal hypertexts. In *Proceedings of the 29th ACM Conference on Hypertext and Social Media*, Proceedings of the 29th ACM Conference on Hypertext and Social Media (HT '18), New York, NY, USA. ACM.

Ines Montani and Matthew Honnibal. 2018. Prodigy: A new annotation tool for radically efficient machine teaching. *Artificial Intelligence*, to appear.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Comput. Linguist.*, 31(1):71106.

Thies Pfeiffer. 2012. Using virtual reality technology in linguistic research. In *2012 IEEE Virtual Reality Workshops*, VRW '12, pages 83–84.

Christian Plotzky, Ulrike Lindwedel, Michaela Sorber, Barbara Loessl, Peter Knig, Christophe Kunze, Christiane Kugler, and Michael Meng. 2021. Virtual reality simulations in nurse education: A systematic mapping review. *Nurse Education Today*, 101.

James Pustejovsky. 2017. ISO-TimeML and the annotation of temporal information. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 941–968. Springer Netherlands, Dordrecht.

James Pustejovsky, Jessica L Moszkowicz, and Marc Verhagen. 2011. ISO-Space: The annotation of spatial information in language. In *Proc. of the Sixth Joint ISO-ACL SIGSEM Workshop on ISA*, pages 1–9.

James Pustejovsky and Amber Stubbs. 2012. *Natural Language Annotation for Machine Learning*. O'Reilly, Sebastopol, CA.

Anna Rumshisky and Amber Stubbs. 2017. Machine learning for higher-level linguistic tasks. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 333–351. Springer Netherlands, Dordrecht.

Helmut Schumacher, Jacqueline Kubczak, Renate Schmidt, and Vera de Ruiter. 2004. *VALBU - Valenzwörterbuch deutscher Verben*. Narr, Tübingen.

Burr Settles. 2012. Active learning. *Synthesis lectures on artificial intelligence and machine learning*, 6(1):1–114.

Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, ANLP-97, Washington, DC.

Alen Smaji. 2020. Development and testing of an interactive 3d city model using the example of the local public transport network of the city of frankfurt. bachelor's thesis, Institute of Computer Science and Mathematics, Text Technology Lab, Johann Wolfgang Goethe-Universitt, Frankfurt, Germany. Original title: Entwicklung und Erprobung eines interaktiven 3D-Stadtmodells am Beispiel des Personennahverkehrsnetzwerks der Stadt Frankfurt.

Christian Spiekermann, Giuseppe Abrami, and Alexander Mehler. 2018. VAnnotatoR: a gesture-driven annotation framework for linguistic and multimodal annotation. In *Proceedings of the Annotation, Recognition and Evaluation of Actions (AREA 2018) Workshop*, AREA.

S. Wagner. 2016. Natural language processing is no free lunch. In Tim Menzies, Laurie Williams, and Thomas Zimmermann, editors, *Perspectives on Data Science for Software Engineering*, pages 175–179. Morgan Kaufmann, Boston.

Graham Wilcock. 2017. The evolution of text annotation frameworks. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 193–207. Springer Netherlands, Dordrecht.

Jason Wither, Stephen DiVerdi, and Tobias Höllerer. 2009. Annotation in outdoor augmented reality. *Computers & Graphics*, 33(6):679–689.

Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. 2006. ELAN: A professional framework for multimodality research. In *In Proceedings of the 5th International Conference on Language Resources and Evaluation*, LREC 2006, pages 1556–1559.