The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation

Jan-Christoph Klie Michael Bugert Beto Boullosa
Richard Eckart de Castilho Iryna Gurevych
Ubiquitous Knowledge Processing Lab, Technische Universität Darmstadt, Germany
`https://www.ukp.tu-darmstadt.de`

Abstract

We introduce INCEpTION, a new annotation platform for tasks including interactive and semantic annotation (e.g., concept linking, fact linking, knowledge base population, semantic frame annotation). These tasks are very time consuming and demanding for annotators, especially when knowledge bases are used. We address these issues by developing an annotation platform that incorporates machine learning capabilities which actively assist and guide annotators. The platform is both generic and modular. It targets a range of research domains in need of semantic annotation, such as digital humanities, bioinformatics, or linguistics. INCEpTION is publicly available as open-source software.[1]

1 Introduction

Due to the success of natural language processing (NLP), there is a large interest to apply NLP methods in a wide range of new application domains, for instance to scale textual data analysis or to explore textual data. This requires being able to quickly bootstrap new annotated corpora in these domains. As target users, we consider for instance data scientists who train and evaluate machine learning algorithms as well as researchers who to wish to cross-reference and disambiguate text collections for better exploration and discovery. Furthermore, every application domain uses specific semantics and vocabularies which need to be modeled, making entity linking one of the most important annotation tasks. Thus, we identify three requirements that annotation tools must meet in order to address today's demands:

Annotation assistance. Creating annotated corpora is challenging and requires experts who are highly familiar with the annotation schemes in order to reach high inter-annotator agreement as well as high quality annotations. For semantic annotations, it is even more difficult: tasks such as entity and fact linking are very time intensive and often require an in-depth familiarity with the inventory of the resource. To improve the efficiency of these tasks, it is necessary to create an environment in which the computer can learn from the human and use this knowledge to support the human annotator.

Knowledge management. Semantic resources for new domains typically do not exist from the start. Instead, they are constructed and expanded as part of the annotation task. Thus, while some annotation tools already support entity linking against existing large-scale general knowledge bases (KB) such as Wikidata or DBPedia, it is also necessary that domain specific knowledge can be collected and modeled directly in the annotation tool.

Customizability and extensibility. Every annotation project has specific requirements that go beyond the basic task requirements, e.g. due to the data formats, knowledge resources, or text genres involved. Therefore, it is important that the tool can be customized, extended, and adapted to novel tasks.

INCEpTION addresses these requirements in several ways. To improve the efficiency of (semantic) annotation tasks, so-called *recommenders* are implemented which provide users with suggestions for possible labels. To navigate the annotation suggestions, an *active learning mode* can be enabled which

[1] `https://inception-project.github.io` ; software is licensed under the Apache License 2.0

Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations, pages 5–9
Santa Fe, New Mexico, USA, August 20-26, 2018.

guides the annotator in an efficient and effective manner. *Knowledge management* is fully integrated; knowledge bases can be created and edited, *entity and fact linking* is supported. The *modular architecture* of INCEpTION enables users to augment their instance with custom machine learning algorithms, data formats, knowledge bases, annotation types, visualizations and more.

2 Related Work

In recent years, several knowledge management and annotation tools have been developed, but none of them offer an integrated environment addressing all of the mentioned requirements.

Several tools, e.g. GATE Teamware (Bontcheva et al., 2013) implement support for automatically pre-annotating text. These are then corrected by the annotator in the next step. In contrast to that, INCEpTION allows recommenders to give suggestions at any time during the annotation process and learns from the user interactions (new annotations, rejections, etc.).

WebAnno (Yimam et al., 2014) integrates an automation mode in which the system can learn from annotations made by the user and provide suggestions. However, retraining has to be triggered manually by an administrator. Also, it uses a non-modular backend that provides only one machine learning algorithm and does not support active learning. WebAnno presents the document to be annotated and the recommended annotations separately in a split-screen mode which makes it tedious to relate recommendations to already existing annotations.

The general approach described by Emanuele Pianta and Zanoli (2008) who integrate an active learning process with an existing annotation tool and the ability to call out to different machine learning backends for recommendations as well as *Prodi.gy*[2] are similar to our approach. However, they focus strongly on the active learning aspect and force the user to follow the lead of the active learning module, restricting the user's workflow. In INCEpTION, the active learning algorithm highlights a particular recommendation to be judged by the user, but does not prevent the user from making other annotations.

The web-based tool AlvisAE (Papazian et al., 2012) supports both linking entity mentions to a knowledge base and editing knowledge bases (with limitations), but it does not support recommendations or active learning. Knowtator (Ogren, 2006) is another instance of a desktop application which ships as an annotation plugin for an ontology building tool. However, single-user tools like the ones above do not meet today's demand for collaboration-oriented annotation tools.

3 INCEpTION – System Overview

INCEpTION offers a number of functionalities expected from a generic annotation platform: a versatile and yet intuitive user interface, flexible configuration of the annotation schema, the ability to run multiple annotation projects concurrently for multiple users and workflow-support with annotation and adjudication stages, etc. With respect to these basic functionalities, we build on our previous work in the context of WebAnno (Yimam et al., 2014) and UIMA (Ferrucci et al., 2009), and therefore refer the interested readers to these projects.

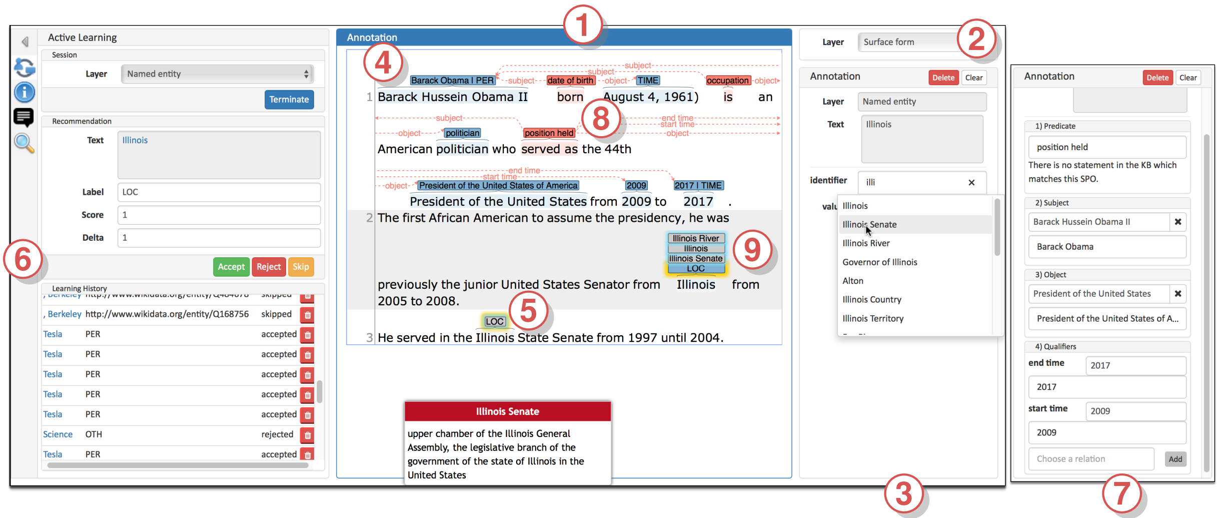
In this paper, we focus specifically on INCEpTION's unique features, in particular on annotation assistance via *recommenders* and *active learning*, the *knowledge management capabilities* and its options for *customizations and extensions*.

Annotation User Interface The annotation scheme used by INCEpTION organizes annotations into layers which define the set of attributes that an annotation may carry. Users can define an arbitrary number of layers that are each either spans or relations between spans. Each layer can have an arbitrary number of features which can be strings, numbers booleans, concept references, or references to other annotations.

The annotation user interface (Figure 1) displays the document text in the central part ①. Marking a span of text here creates a new annotation on the layer that is selected in the right sidebar ② (e.g. named entity). Span annotations are displayed as bubbles above the text.

When an annotation is created or an existing annotation is selected, its features are shown in the right sidebar and can be edited there ③④. Depending on the feature type, a specialized editor is shown. For

[2] https://prodi.gy



Layer: Named entity

Text: Illinois

Label: LOC

Score: 1

Delta: 1

, Berkeley	http://www.wikidata.org/entity/Q484078	skipped	
, Berkeley	http://www.wikidata.org/entity/Q168756	skipped	
Tesla	PER		accepted
Tesla	PER		accepted
Tesla	PER		accepted
Tesla	PER		accepted
Tesla	PER		accepted
Science	OTH		rejected
Tesla	PER		accepted

1 Barack Hussein Obama II born August 4, 1961) is an American politician who served as the 44th President of the United States from 2009 to 2017 .

2 The first African American to assume the presidency, he was previously the junior United States Senator from Illinois from 2005 to 2008.

3 He served in the Illinois State Senate from 1997 until 2004.

Illinois Senate

upper chamber of the Illinois General Assembly, the legislative branch of the government of the state of Illinois in the United States

Layer: Surface form

Layer: Named entity

Text: Illinois

identifier: illi

value: Illinois, Illinois Senate, Illinois River, Governor of Illinois, Alton, Illinois Country, Illinois Territory

1) Predicate: position held

There is no statement in the KB which matches this SPO.

2) Subject: Barack Hussein Obama II — Barack Obama

3) Object: President of the United States — President of the United States of A...

4) Qualifiers

end time: 2017 — 2017

start time: 2009 — 2009

Choose a relation

Figure 1: INCEpTION annotation editor: (1) annotation area, (2) annotation layer selection, (3) entity linking feature editor, (4) named entity linked to Wikidata, (5) entity mention suggestion, (6) active learning sidebar, (7) fact linking editor, (8) annotated fact, (9) entity linking recommendations.

instance, the editor to assign concepts from a knowledge base is an auto-complete input field which shows entities from the knowledge base that match the users' input. The left sidebar provides access to further functionalities, in particular to the active learning mode.

Recommenders To improve annotation efficiency, INCEpTION offers *recommenders*. These are algorithms that make use of machine learning and/or knowledge resources to provide annotation suggestions; they are displayed to the user alongside already made annotations in a different color (5). The user may accept a suggestion by clicking on it. This turns the suggestion into a proper annotation which can then be further edited if desired. The user may also reject the suggestion by double-clicking on it.

The recommender subsystem is designed to continuously monitor the users' actions, to update/retrain the recommendation models, and to provide always up-to-date suggestions. Multiple recommenders can be used simultaneously, e.g. high-precision/low-recall recommenders (e.g. using a dynamic dictionary) which are useful during early annotation stages, and context sensitive recall-oriented classifiers (e.g. sequence classifiers) for later stages. To avoid classifiers providing too many wrong suggestions during bootstrapping, a quality threshold can be configured per recommender.

INCEpTION supports two types of recommenders: internal and external. Internal recommenders are directly integrated into the platform by implementing a Java interface, while external recommenders use a simple, HTTP-based protocol to exchange UIMA CAS XMI (a XML representation of UIMA annotations). External recommenders allow users to leverage already existing and pre-trained machine learning models or libraries from other programming languages.

Active learning The goal of active learning (AL) is to quickly reach a good quality of annotation suggestions by soliciting feedback from the user that is expected to be most informative to the underlying machine learning algorithm. Presently, we use the uncertainty sampling strategy (Lewis and Gale, 1994) to drive the AL as it only requires that the recommenders produce a confidence score for each suggestion. The AL mode (6) works for one layer at a time to avoid confusion. After the layer has been selected, the system highlights the suggestion it seeks input for in the annotation area and displays its details in the AL sidebar. The user can then accept, reject or skip the suggestion. Skipped suggestions are presented again to the user when there are no more suggestions to accept or reject. The choices are stored in the learning history where the user can review and undo them if necessary. When the AL mode is enabled, the user can still deviate from its guidance and arbitrarily create and modify annotations. All changes made through the AL sidebar or in the main editor are immediately picked up by the recommenders causing the suggestions as well as the AL guidance to be updated.

Knowledge Management For knowledge management, INCEpTION supports RDF-based knowledge bases. While internal KBs can be used for small domain-specific knowledge, large external (remote) knowledge bases can be accessed via SPARQL. A flexible configuration mechanism is used to support different knowledge representations, such as Wikidata, DBPedia, OWL, CIDOC-CRM, SKOS, etc. INCEpTION has notions of classes, instances, properties and qualifiers (for KBs using reification). However, it does not aspire to offer full support for advanced features of schemes such as OWL.

Knowledge bases enable the user to perform knowledge-driven annotations, e.g. annotating mentions of knowledge base entities in documents (entity linking ③) or creating new knowledge bases by annotating subjects, predicates and objects in text (fact linking ⑦⑧). Users can also explore and edit the knowledge base contents within INCEpTION.

To facilitate the entity linking process, INCEpTION can optionally take into account the context of the entity mention in order to provide the user with a ranked list of potential candidates. The same approach is used to drive an entity linking recommender which displays high-ranking candidates as annotation suggestions ⑨ in the annotation area where the user can accept them with a single click.

Customizability and extensibility There are two approaches to customize and extend INCEpTION:

Internal extensions. The dependency injection and event mechanisms of Spring Boot[3] are used to internally modularize INCEpTION. Extension points make it possible to register new types of annotation properties, new editors or new internal recommenders. Modules can coordinate their tasks with each other through events. As an example, the main annotation area issues an event when an annotation has been created or changed. The recommenders and the AL mode react to this event in order to update themselves. Functionality can thereby not only be added but also removed to create custom branded versions of INCEpTION. The event-driven modular approach also enables the system to comprehensively log user and system actions. This data can for instance be used by annotation project managers in order to evaluate the performance of their annotators.

External extensions. Currently supported are external recommenders and knowledge bases. Benefits of using external services include increased stability (failing services do not crash the entire platform), scalability (deploy resource-hungry services on different machines) and the free choice of programming language (e.g. most deep learning frameworks are not implemented in Java).

Additionally, INCEpTION uses (de-facto) standards such as UIMA for annotations and RDF, OWL and SPARQL for knowledge bases to achieve a high level of interoperability with existing tools and resources.

4 Use cases

To ensure that INCEpTION remains generic, we collaborate with multiple use cases:

FAMULUS. Schulz et al. (2017) use INCEpTION to annotate medical case study reports with argument components. These are used to train a machine learning model which evaluates the diagnostic competence of aspiring doctors. A pre-trained deep learning model is integrated as an external recommender and is used during annotation. The annotators that use INCEpTION in conjunction with the recommenders report the usefulness and improvement in annotation speed and quality.

EDoHa. Stahlhut et al. (2018) have created a hypothesis validation tool using INCEpTION. It features a hypothesis/evidence editor which allows users to create hypotheses and link evidence in the form of text paragraphs to it.

Knowledge-driven rntity ranking. In order to support users during entity linking, the re-ranking approach described in (Sorokin and Gurevych, 2018) was adapted and integrated into INCEpTION. It is used as a recommender and in the auto-suggestion box for the named entity layer.

As part of the collaborations with the above use-cases, INCEpTION logs the users' actions in order to investigate for instance which assistive features (i.e. recommenders) work best for the respective tasks, whether they introduce a bias in the annotator's results, and how to improve the user interface for an improved user experience.

[3] Spring Boot: `https://projects.spring.io/spring-boot/`

5 Conclusion and Future Work

In this paper, we have presented INCEpTION which –to the best of our knowledge– is the first modular annotation platform which seamlessly incorporates recommendations, active learning, entity linking and knowledge management. Our approach provides a number of advantages over current state-of-the-art annotation tools. Recommenders giving suggestions on-line allow users to annotate texts more quickly and accurately. External recommenders can be added to leverage already existing machine learning models and bootstrap the annotation for new domains. Knowledge management is directly integrated which allows entity- and fact linking together with building the knowledge base on the fly. The modular approach used by INCEpTION provides users with the possibility to tailor the platform according to their needs, for instance by adding new machine learning algorithms, annotation editors or knowledge bases.

INCEpTION is publicly available as open source-software. We welcome early adopters and encourage feedback for a continued alignment of the platform with the needs of the community. Several collaborations are on the way to develop and improve features that are useful to researchers and annotators, e.g. corpus search, recommenders that check the plausibility of annotations or fully custom user interfaces.

Acknowledgements

We thank Wei Ding, Peter Jiang and Marcel de Boer and Naveen Kumar for their valuable contributions and Teresa Botschen and Yevgeniy Puzikov for their helpful comments. This work was supported by the German Research Foundation under grant No. EC 503/1-1 and GU 798/21-1 (INCEpTION).

References

Kalina Bontcheva, Hamish Cunningham, Ian Roberts, Angus Roberts, Valentin Tablan, Niraj Aswani, and Genevieve Gorrell. 2013. GATE Teamware: A Web-based Collaborative Text Annotation Framework. *Language Resources and Evaluation*, 47(4):1007–1029.

Christian Girardi Emanuele Pianta and Roberto Zanoli. 2008. The TextPro Tool Suite. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 2603–2607.

David Ferrucci, Adam Lally, Karin Verspoor, and Eric Nyberg. 2009. Unstructured information management architecture (UIMA) version 1.0. OASIS Standard.

David D. Lewis and William A. Gale. 1994. A Sequential Algorithm for Training Text Classifiers. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12.

Philip V. Ogren. 2006. Knowtator: A Protégé plug-in for annotated corpus construction. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Demonstrations*, pages 273–275.

Frédéric Papazian, Robert Bossy, and Claire Nédellec. 2012. AlvisAE: a collaborative Web text annotation editor for knowledge acquisition. In *Proceedings of the Sixth Linguistic Annotation Workshop*, pages 149–152. Association for Computational Linguistics.

Claudia Schulz, Michael Sailer, Jan Kiesewetter, Christian M. Meyer, Iryna Gurevych, Frank Fischer, and Martin R. Fischer. 2017. Fallsimulationen und automatisches adaptives Feedback mittels Künstlicher Intelligenz in digitalen Lernumgebungen. *e-teaching.org Themenspecial "Was macht Lernen mit digitalen Medien erfolgreich?"*, pages 1–14.

Daniil Sorokin and Iryna Gurevych. 2018. Mixing context granularities for improved entity linking on question answering data across entity categories. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 65–75.

Chris Stahlhut, Christian Stab, and Iryna Gurevych. 2018. Pilot experiments of hypothesis validation through evidence detection for historians. In *Proceedings of Design of Experimental Search & Information REtrieval Systems (DESIRES)*. (in press).

Seid Muhie Yimam, Chris Biemann, Richard Eckart de Castilho, and Iryna Gurevych. 2014. Automatic Annotation Suggestions and Custom Annotation Layers in WebAnno. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 91–96.