INCEpTION
CORPORUS-BASED DATA SCIENCE FROM SCRATCH

Richard Eckart de Castilho, Jan-Christoph Klie, Naveen Kumar, Beto Boulosa, Iryna Gurevych

Ubiquitous Knowledge Processing Lab (UKP-TUDA)
Dept. of Computer Science
Technische Universität Darmstadt

Motivation

- Corpus-based data science is seeing rapid adoption in science and industry
- Domain-specific annotated corpora are required in many domains
- Current corpus processes for manual corpus annotation do not scale

INCEpTION is an infrastructure-ready human-in-the-loop annotation platform combining machine learning and human expertise for rapid domain adaptation

INCEpTION Platform

INCEpTION aims to support three functionalities which are commonly required for text annotation projects but typically not available in a single tool: corpus creation, text annotation, knowledge management.

The platform additionally provides assistive features such as machine learning recommenders to help users working on these tasks more efficiently.

Integrating these functionalities into a single comprehensive platform permits addressing tasks typically not found in generic annotation platforms, such as entity linking, knowledge base population, cross-document coreference annotation, etc.

Open Development

While most annotation tools are built in annotation projects, INCEpTION is an infrastructure software project and is not associated with any single annotation project. Acquiring early adopters and aligning with their use-cases is a key part of our mission.

This motivates our open development philosophy:

- All code is open and publicly available on GitHub under the liberal Apache License 2.0
- All development-related tasks and issues are publicly managed and discussed via GitHub
- Internal and private communication is kept at a minimum

INCEpTION is an infrastructure-ready human-in-the-loop annotation platform combining machine learning and human expertise for rapid domain adaptation

INCEpTION Platform

INCEpTION aims to support three functionalities which are commonly required for text annotation projects but typically not available in a single tool: corpus creation, text annotation, knowledge management.

The platform additionally provides assistive features such as machine learning recommenders to help users working on these tasks more efficiently.

Integrating these functionalities into a single comprehensive platform permits addressing tasks typically not found in generic annotation platforms, such as entity linking, knowledge base population, cross-document coreference annotation, etc.

Open Development

While most annotation tools are built in annotation projects, INCEpTION is an infrastructure software project and is not associated with any single annotation project. Acquiring early adopters and aligning with their use-cases is a key part of our mission.

This motivates our open development philosophy:

- All code is open and publicly available on GitHub under the liberal Apache License 2.0
- All development-related tasks and issues are publicly managed and discussed via GitHub
- Internal and private communication is kept at a minimum

Project Workflow

INCEpTION is an infrastructure-ready human-in-the-loop annotation platform combining machine learning and human expertise for rapid domain adaptation

Knowledge Bases

Within the NLP and text mining landscape, an annotation platform like INCEpTION only covers a part of the overall text analysis needs. Therefore, it is important that the platform is open and interoperable with external services and resources.

Knowledge Resources

- RDF-based knowledge resources are supported
- Knowledge bases are supported
- Knowledge bases are supported
- Support for feature detection and knowledge bases (Slovakian word supervision)

Intra-domain and inter-domain linking

- Entity and fact linking
- Linking of domains and resources
- Linking of entities and resources
- Support for feature detection and knowledge bases (Slovakian word supervision)

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.

Interoperability and Integration

Integration goes beyond interoperability. E.g. when an external text mining platform wants to delegate annotation to the INCEpTION platform, it needs to be able to automatically set up annotation projects, import data, monitor the ongoing annotations and finally retrieve the annotated data for further use.

Extensibility

The architecture modular architecture is realized using the Spring framework. Dependency injection and events are used to achieve a loose coupling between the modules.