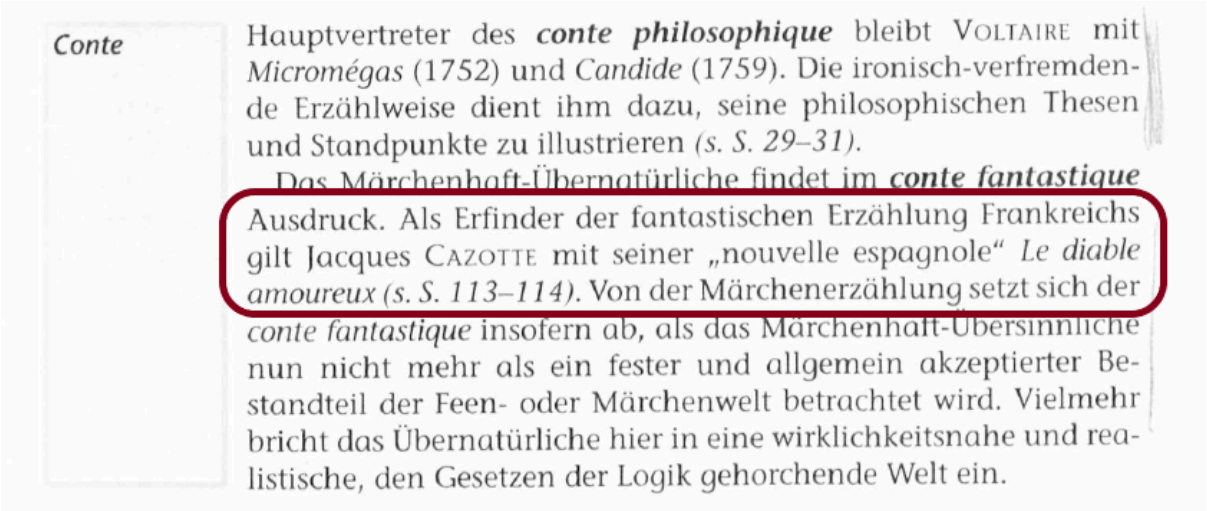
INCEpTION-Workflow

The project MiMoText deals with new ways to model and analyze literary history and literary historiography. We established a knowledge graph called MiMoTextBase[1] which aggregates statements relevant to literary history extracted from three types of sources: bibliographical reference works, primary texts and scholarly publications. Our application domain is the French novel of the second half of the 18th century.

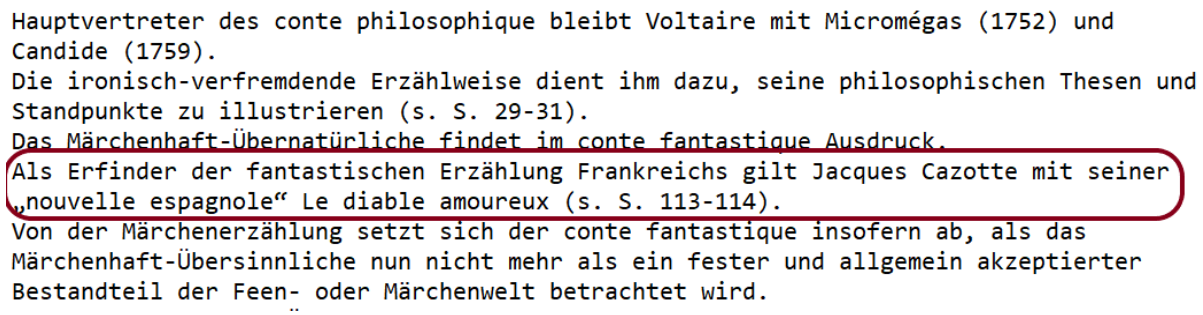
To extract statements on themes, tone and intention from the scholarly literature, the INCEpTION tool was used to manually annotate texts by linking named entities and concepts with entities from Wikidata and the MiMoTextBase and drawing relations between them. A project-specific workflow was developed, starting with the scholarly literature texts as plain text and ending with statements in the form of LOD triples as output, which are imported into the MiMoTextBase.

Preparing and importing the input data

Our input data consisted of scholarly literature on the French novel of the 18th century. For the import into INCEpTION, the texts were scanned, OCRized and converted into plain text format. As we aimed to annotate at sentence level, the files were converted to One-Sentence-Per-Line format with each sentence on its own line. When importing the texts into the INCEpTION project (**Documents** tab on the project **Settings** page) we chose the format “Plain text (one sentence per line)”.



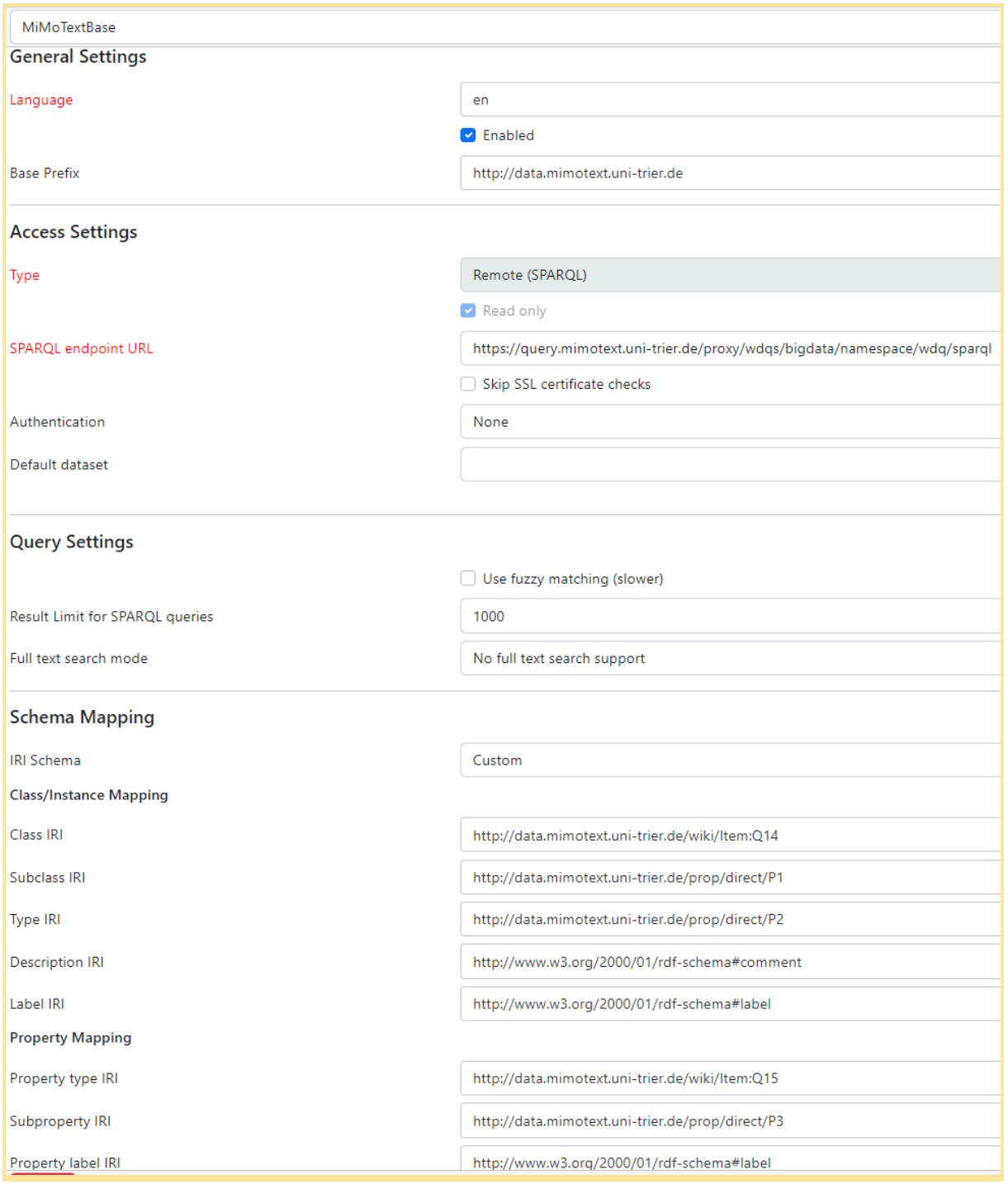
[1] We chose Wikibase (https://wikiba.se/) as an infrastructure for the project, not only because it is a free and open software, but also because it is especially suited for multilingual data. Wikibase is an open source software developed by the Wikimedia Foundation and can be customized to meet specific data management and ontology design needs, making it a flexible and adaptable tool. The integrated Wikidata Query Service (WDQS) provides a SPARQL endpoint and comes along with several built-in visualization options, which facilitate plotting data patterns in various ways, as illustrated in the previous section.



Connecting the MiMoTextBase with INCEpTION and Wikidata

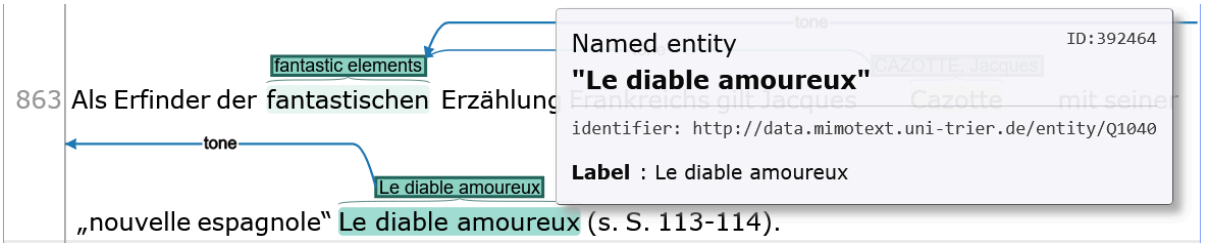
In the MiMoTextBase, all project-related works and authors are stored as items and can be clearly referenced (e.g. Q217, Jacques Cazotte). In addition, various vocabularies were created in the project, such as a thematic or tone vocabulary, whose concepts are also created as entities in the MiMoTextBase and have their own ID (e.g. the tone concept “criticism”, Q3902). The MiMoTextBase contains various properties, such as about (P36) or tone (P38), for describing relationships between items.

INCEpTION was linked to the MiMoTextBase to annotate the “about”, “tone” and “intention” statements. In order to link items that are not included in the MiMoTextBase, we have also used a link to Wikidata as a knowledge base. The possibility to connect knowledge bases in INCEpTION is a great added value that opens up the possibility to transform the results of the annotation process via the workflow documented here. There is already a default setting for connecting an INCEpTION project with Wikidata, which is very useful. By connecting the MiMoTextBase, the exports of annotations could be converted into new statement triples. The SPARQL endpoint of the knowledge base and an IRI schema need to be defined for linking a project-specific knowledge base. If the knowledge base is a Wikibase instance, it is necessary to integrate different standards, as there is no clear separation of classes and instances in the Wikidata / Wikibase data model and as the representation and linking of properties is not very standardized so far. We have therefore taken the IRI schema mapping in INCEpTION into account in the design of our MiMoText ontology which is directly connected to the MiMoText data SPARQL representation.

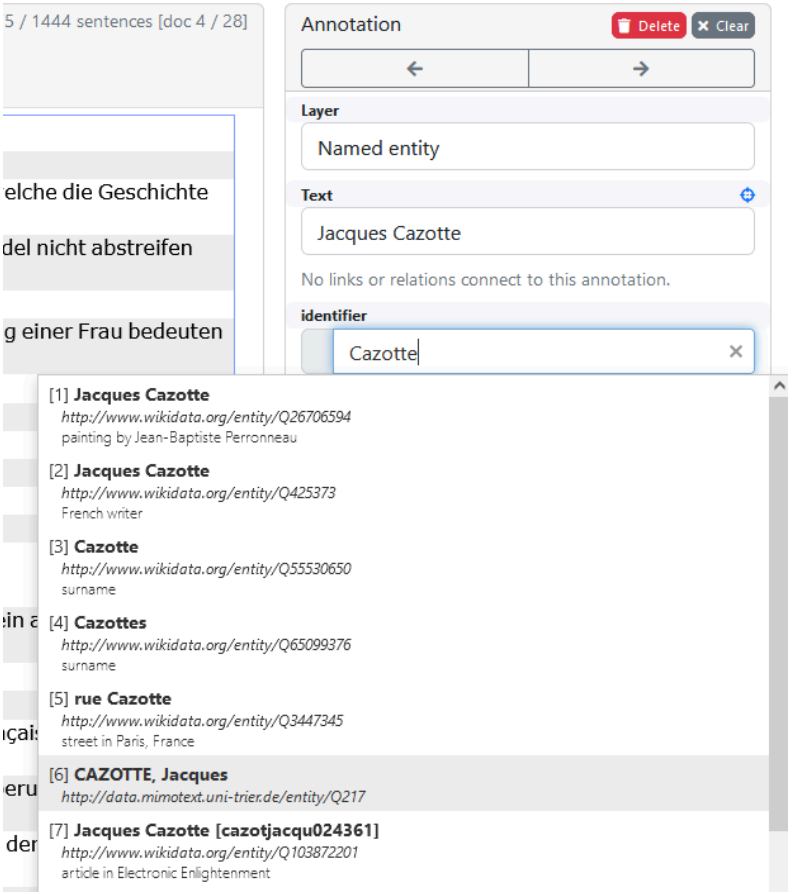


Annotation

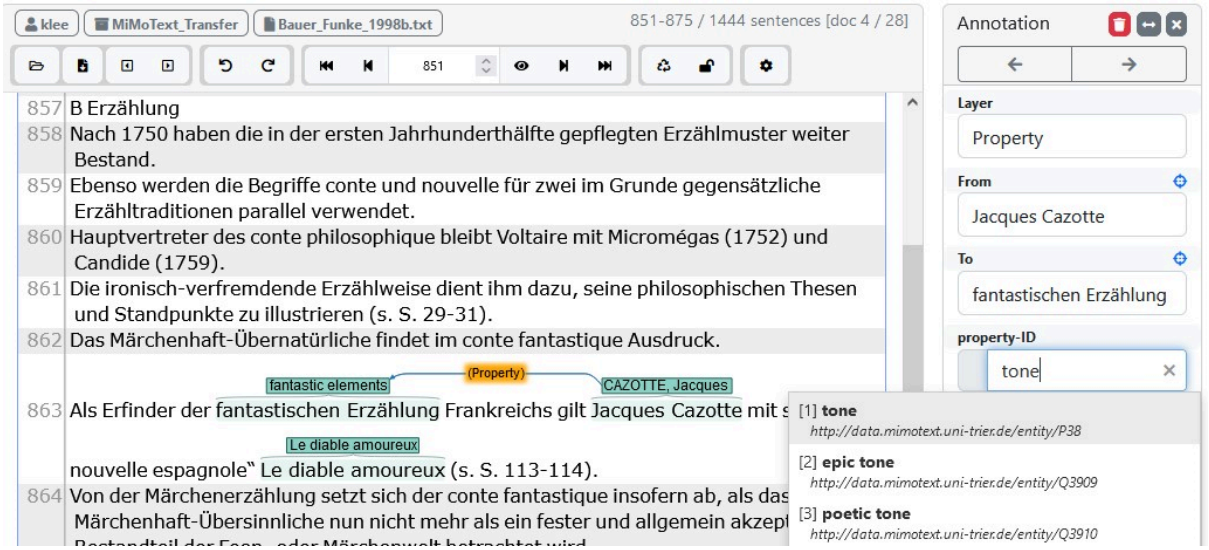
When annotating the texts in INCEpTION, authors, works and mentions of themes, tone or intention were marked as named entities and linked to entities in the MiMoTextBase (or Wikidata) like the work “Le diable amoureux” in the following example:



Because the MiMoTexBase was previously linked to INCEpTION, you can select from a list of suggested matching entities for a string. So when annotating the author Jacques Cazotte, INCEpTION suggests entities from Wikidata and MiMoText. In our case we chose the MiMoText item Q217.



Additionally, relations were drawn between these annotated named entities, each of which was linked to a MiMoText property (cf. P38 "tone" in the example).



Export

For further processing of the annotations, we have exported the texts in **UIMA CAS XMI 1.0** format. It can be processed with the help of Python and the dkpro-cassis library. On this basis, we developed a script with which we extracted the annotations and converted them into statements.

From XMI output to statements

The UIMA CAS XMI structure

- All characters of the annotated text are numbered consecutively.
- An element is created for each sentence.
- Each sentence is given an ID ("xml:id") and is defined by the positions of its first ("begin") and last ("end") character.

```
<type5:Sentence xmi:id="19" sofa="1" begin="0" end="9"/>
<type5:Sentence xmi:id="24" sofa="1" begin="10" end="74"/>
<type5:Sentence xmi:id="29" sofa="1" begin="75" end="174"/>
<type5:Sentence xmi:id="34" sofa="1" begin="175" end="285"/>
<type5:Sentence xmi:id="39" sofa="1" begin="286" end="474"/>
<type5:Sentence xmi:id="44" sofa="1" begin="475" end="627"/>
```

- The same is done for each token of the text:

```
<type5:Token xmi:id="17158" sofa="1" begin="4898" end="4905" order="0"/>
<type5:Token xmi:id="17171" sofa="1" begin="4905" end="4906" order="0"/>
<type5:Token xmi:id="17184" sofa="1" begin="4907" end="4910" order="0"/>
<type5:Token xmi:id="17197" sofa="1" begin="4911" end="4923" order="0"/>
<type5:Token xmi:id="17210" sofa="1" begin="4924" end="4927" order="0"/>
```

- For each string annotated as a named entity, an element is created with an ID ("xml:id"), its position in the text (first and last character - "begin" and "end") and the assigned item from the MiMoTextBase ("identifier"). In our use case, it is a work, an author or a concept.

```
<type4:NamedEntity xmi:id="393891" sofa="1" begin="174795" end="174800"
identifier="http://data.mimotext.uni-trier.de/entity/Q850"/>

<type4:NamedEntity xmi:id="393885" sofa="1" begin="174805" end="174815"
identifier="http://data.mimotext.uni-trier.de/entity/Q3911"/>

<type4:NamedEntity xmi:id="393904" sofa="1" begin="175128" end="175136"
identifier="http://data.mimotext.uni-trier.de/entity/Q842"/>
```

- A property element is created for each relation drawn between two named entities in INCEpTION. It contains the following information:
  - an ID ("xml:id")
  - the span over which the annotation extends (position of the first and last character)
  - the IDs of the two linked named entities ("Dependent" and "Governor")
  - a propertyID with which the relation was linked (corresponds to a property from the MiMoTextBase)

```
<custom:Property xmi:id="390676" sofa="1" begin="36334" end="36346"
   Dependent="390670" Governor="390658"
   propertyID="http://data.mimotext.uni-trier.de/entity/P38"/>
```

Generating statements

Coming back to our Cazotte example: We start with the property element:

```
<custom:Property xmi:id="390676" sofa="1" begin="36334"
   end="36346" Dependent="390670" Governor="390658"
   propertyID="http://data.mimotext.uni-trier.de/entity/P38"/>
```

The **subject** of the statement is the governor of the property. Using its xml:id (`"390658"`), we can access its MiMoText ID (Q1040) in the NamedEntity element,

which represents the work "Le diable amoureux". Using the character positions, we can extract the subject_snippet, the annotated string ("Le diable amoureux"):

```
<type4:NamedEntity xmi:id="390658" sofa="1" begin="36386"
   end="36404"
   identifier="http://data.mimotext.uni-trier.de/entity/Q1040"/>
```

The **object** of the statement is the dependent of the property. Its xml:id (`"390670"`) takes us to its MiMoText ID (Q3899) in the NamedEntity element which is the "fantastic elements" concept from the tone vocabulary. We can use the character positions to extract the object_snippet, the annotated string ("fantastic").

```
<type4:NamedEntity xmi:id="390670" sofa="1" begin="36334"
   end="36346"
   identifier="http://data.mimotext.uni-trier.de/entity/Q3899"/>
```

The **property** of the statement is provided as information in the propertyID attribute: http://data.mimotext.uni-trier.de/entity/P38. This is the MiMoText property "tone" (P38).

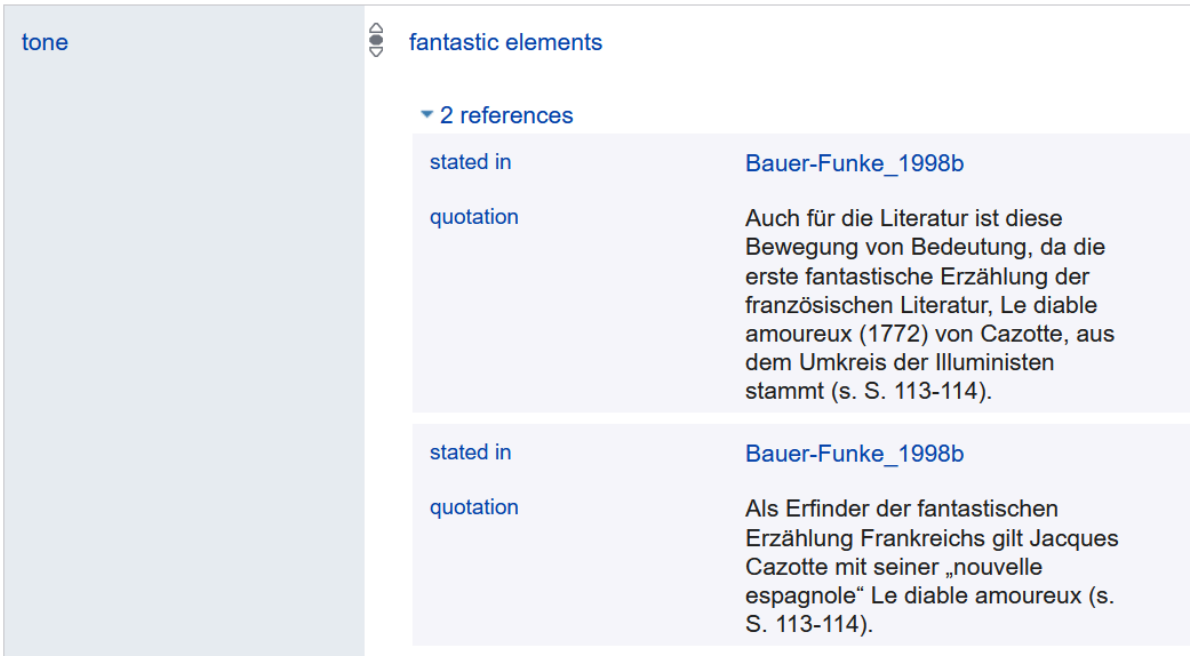
For each statement, the complete sentences over which the associated annotation extends are to be fed in as a quotation. The associated sentences are determined via the positions of the first and last character from the property element and extracted as a coherent string.

In this way, with the help of our script, we obtain all the information we need for our statement:

subject id	subject snippet	property	object id	object label	object snippet	sentence snippet
http://data.mimotext.uni-trier.de/entity/Q1040	Le diable amoureux	http://data.mimotext.uni-trier.de/entity/P38 (tone)	http://data.mimotext.uni-trier.de/entity/Q3899	fantastic elements	fantastischen	Als Erfinder der fantastischen Erzählung Frankreichs gilt Jacques Cazotte mit seiner „nouvelle espagnole“ Le diable amoureux (s. S. 113-114).

To import new statements into our MiMoTextBase, we developed an adapted PyWikibot for automated imports based on tsv files.[2]

And this is how the statement finally appears in the MiMoTextBase:



In our tutorial, we have a section that specifically demonstrates queries for statements based on the scholarly literature annotated in INCEpTION.

[2] See the repository for our customized WikibaseBot: https://github.com/MiMoText/wikibase-bot. In addition, we also used the tool “QuickStatements” written by Magnus Manske, see: https://www.wikidata.org/wiki/Help:QuickStatements.