Annotation Scheme

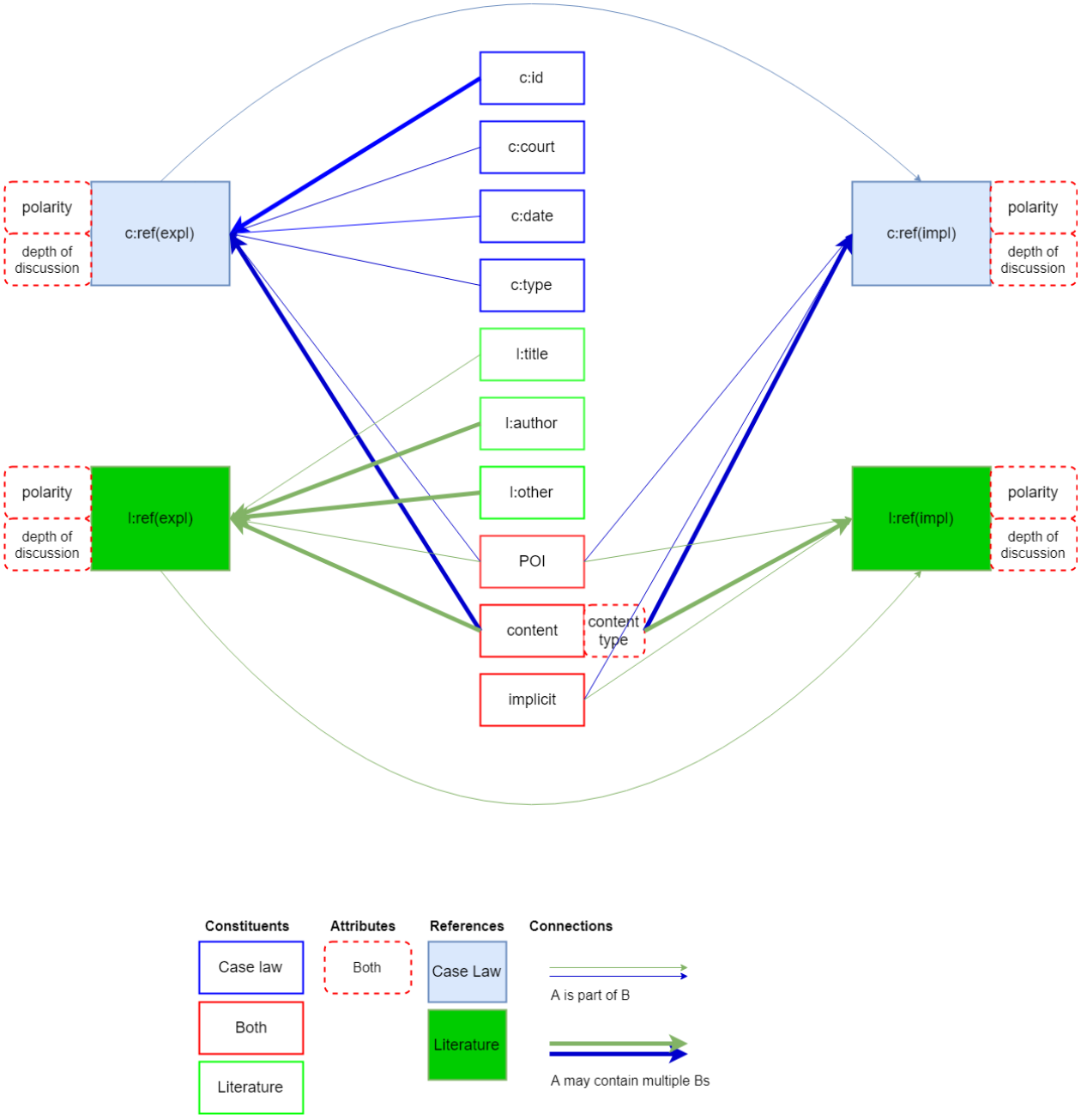
The references in the dataset point either to case law (blue rectangles in the upper part of the diagram below) or to literature (larger green rectangles in the lower part). Moreover, references are either explicit (left side of the diagram) or implicit (right side of the diagram). Every annotated reference consists of basic units and, in case of the implicit references, of a link to the related explicit reference.

First type of basic units is so called *constituents*. These categorize individual annotated spans of text. Constituents are either related to references to case law (blue rectangles in the middle of the diagram below), literature (green rectangles) or to both types of references (red rectangles). Second type of basic units is so called *attributes* (rectangles with interrupted red frame). These are related to either a constituent (content) or the annotated reference as a whole and present a specific value that provides additional information about the reference or its content (polarity, depth of discussion and type of the content).

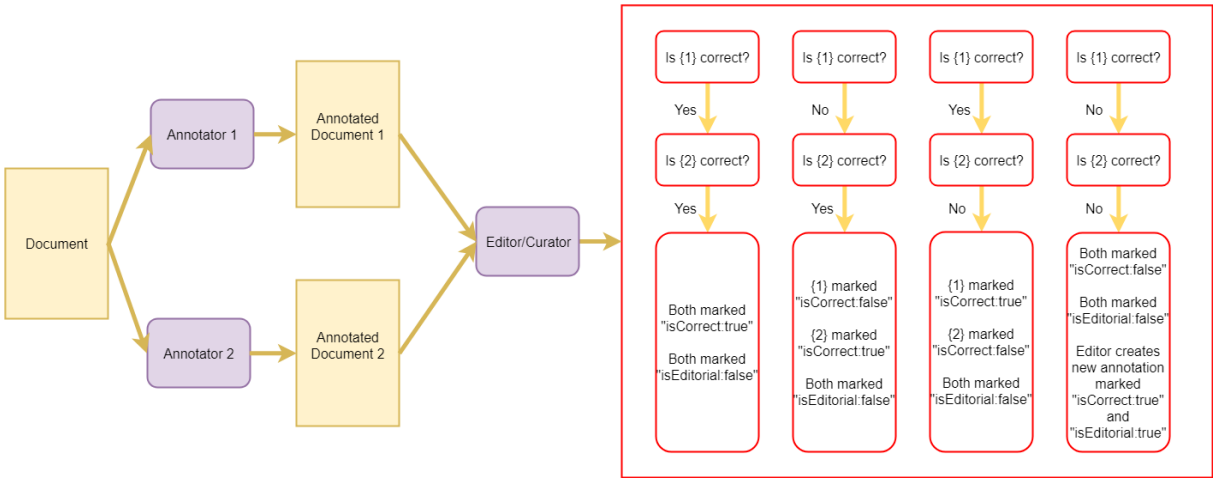
Arrows in the diagram indicate how each annotated reference is put together from different basic units.

Examples:

1) The *c:id* constituent, which identifies the referred case, is a basic unit representing the corresponding annotated text span containing this information.
2) The constituent c*ontent* is an annotated text span indicating why the specific reference to case law or literature is being made. This constituent is directly tied to the attribute *content type*. A single annotated reference may contain multiple *content* constituents with different *content type* attributes.
3) Every explicit case law reference *c:ref(expl)* requires to be assigned several constituents and attributes in order to be constructed and identified properly. As indicated in the diagram, it requires one or multiple *c:id* constituents, one *c:court* constituent, one *c:date* constituent, one *c:type* constituent, one *POI* constituent, one or multiple *content* constituents (with a set value of the attribute *content type*), and also set values of the attributes *polarity* and *depth of discussion*.
4) Similarly, every implicit literature reference *l:ref(impl)* requires to be assigned a different set of constituents and attributes in order to be constructed and identified properly. As shown in the diagram, it requires one *POI* constituent, one or multiple *content* constituents (with a set value of the attribute *content type*), one *implicit* constituent and also set values of the attributes *polarity* and *depth of discussion*. Most importantly, as it is an implicit reference, it requires a set link to one explicit literature reference *l:ref(expl)*, which allows to understand which previous reference is implied in the implicit reference.


c:id
c:court
c:date
c:type
l:title
l:author
l:other
POI
content
implicit
content type
polarity
depth of discussion
c:ref(expl)
l:ref(expl)
c:ref(impl)
l:ref(impl)
Constituents
Case law
Both
Literature
Attributes
Both
References
Case Law
Literature
Connections
A is part of B
A may contain multiple Bs

Annotation Pipeline



Two annotators have annotated every document in the dataset. These annotators were not familiar with each other's work. Every document annotated by two annotators has been passed to editor/curator, whose task was to compare the annotated documents, and mark every annotated reference, constituent and attribute as either correct or incorrect. If editor/curator marked work of both annotators as incorrect, he was then required to re-annotate the reference, and therefore to create correct editorial annotations. Raw corpus contains all annotations (correct, incorrect and created by editor). Gold corpus contains only correct annotations (either marked as correct or created by editor).

Constituents and References (EN to CZ)

Diagram on page 2 is in English, however name convention used in corpus is in Czech. To account for this, we provide comparison of English (used throughout this ReadMe) and Czech (used in corpus) designations.

Constituents	
c:id	J:Identifikátor
c:court	J:Soud
c:date	J:Datum
c:type	J:Druh
l:author	L:Autor
l:title	L:Název
l:other	L:Další údaje
POI	Element
content	Argument
content type	Druh argumentu
implicit	ImplicitníIdentifikace
polarity	Sentiment
depth of discussion	Koeficient
References	
c:ref(expl)	Reference-Judikatura
l:ref(expl)	Reference-Literatura
c:ref(impl)	Reference-Judikatura-implicitní
l:ref(impl)	Reference-Judikatura-explicitní

Corpus File Structure

```
[
    { "txt": >>text of a court decision<<,
      "constituents": [
      { "id": >>unique id that can be used to tie the constituent to a reference<<,
        "start": >>starting offset in the text<<,
        "end": >>ending offset in the text<<,
        "type": >>type of the constituent<<,
        "attributes": [
        { >>attribute's name<<: >>attribute's value<<
        } ...
        ],
        "isCorrect": >>true if marked correct by an editor, false otherwise<<,
        "isEditorial": >>true if created by an editor, false otherwise<<,
        "author": >>author's id<<
      } ...
      ],
      "references": [
      { "id": >>unique id that can be used to tie the constituent to a reference<<,
        "label": >>user friendly label<<,
        "type": >>type of the reference<<,
        "attributes": [
        { >>attribute's name<<: >>attribute's value<<
        } ...
        ],
        "isCorrect": >>true if marked correct by an editor, false otherwise<<,
        "isEditorial": >>true if created by an editor, false otherwise<<,
        "author": >>author's id<<
      } ...
      {
      }
      ]
    }
    ...
]
```

Examples (in Python)

1. Load Documents.

```
with open('corpus.json', 'r') as f:
    docs = json.load(f)
```

2. Print all c:id constituents from the first document.

```
doc = docs[0]
txt = doc['txt']
c_id_constituents = [c for c in doc['constituents']
                     if c['type'] == 'J:Identifikátor']
for c_id in c_id_constituents:
    start = c_id['start']
    end = c_id['end']
    print(txt[start:end])
```

3. Count all c:id constituents across the corpus.

```
count_c_id = 0
for doc in docs:
    c_id_constituents = [c for c in doc['constituents']
                         if c['type'] == 'J:Identifikátor']
    count_c_id += len(c_id_constituents)
```

Suggested Citation

HARAŠTA, Jakub, Jaromír ŠAVELKA, František KASL, Adéla KOTKOVÁ, Pavel LOUTOCKÝ, Jakub MÍŠEK, Daniela PROCHÁZKOVÁ, Helena PULLMANNOVÁ, Petr SEMENIŠÍN, Tamara ŠEJNOVÁ, Nikola ŠIMKOVÁ, Michal VOSINEK, Lucie ZAVADILOVÁ a Jan ZIBNER. Annotated Corpus of Czech Case Law for Reference Recognition Tasks. In Sojka, P., Horák, A., Kopeček, I., Pala, K.. *Text, Speech, and Dialogue: 21st International Conference*. Cham: Springer Nature Switzerland AG, 2018. s. 239-250, 12 s. ISBN 978-3-030-00793-5. doi:10.1007/978-3-030-00794-2_26.

Acknowledgment

Supported by the Czech Science Foundation under grant no. GA17-20645S.