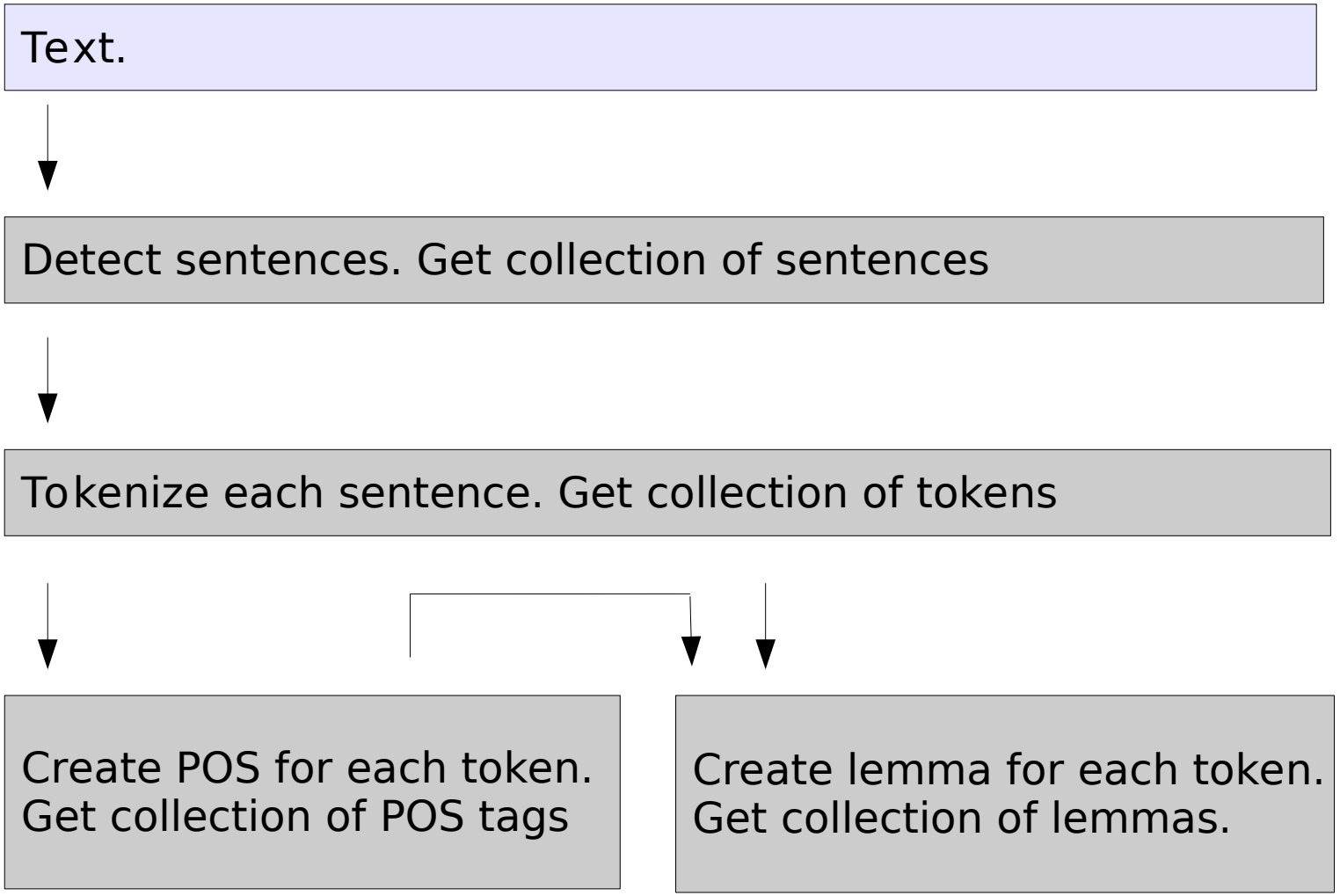
Open NLP

- Java NLP library
- APIs for tokenization,POS tagging, lemmatization, language detection...
- Each processing step needs a model which controls the step. For every language and each processing step a separate model is needed.

`https://opennlp.apache.org/models.html`

OpenNLP – Precessing schema



OpenNLP – Sentence detection

- Load model
- Create sentence detector on model
- Detect sentences

```
try (InputStream modelIn = new FileInputStream("de-sent.bin")) {
   SentenceModel model = new SentenceModel(modelIn);
   SentenceDetectorME sentenceDetector = new SentenceDetectorME(model);
   this.sentences = sentenceDetector.sentDetect(this.text);
} catch (IOException e) {
    e.printStackTrace();
}
```

OpenNLP - Tokenization

- Load model
- Create Tokenizer
- Tokenize sentence

```
try (InputStream modelIn = new FileInputStream("de-
token.bin")) {
  TokenizerModel model = new TokenizerModel(modelIn);
  Tokenizer tokenizer = new TokenizerME(model);
  for (String s : sentences) {
     String[] sTokens = tokenizer.tokenize(s);
  }
} catch (IOException e) {e.printStackTrace();}
```

OpenNLP – POS tagging

```
try (InputStream modelIn = new
FileInputStream("de-pos-maxent.bin")) {
    POSModel model = new POSModel(modelIn);
    POSTaggerME tagger = new POSTaggerME(model);
    for (String[] st : tokens) {
       String[] tags = tagger.tag(st);
    }
} catch (IOException e) {
  e.printStackTrace();
}
```

OpenNLP - Lemmatizer

Lemmatizer needs arrays of words **and** the respective POS tags

---

```
try (InputStream modelIn = new FileInputStream("de-lemmatizer.bin"))
{
   LemmatizerModel model = new LemmatizerModel(modelIn);
   LemmatizerME lemmatizer = new LemmatizerME(model);

   for (int i = 0; i<tokens.size(); i++ )   {
     List<String> st = tokens.get(i);
     List<String> tmpPos = posTags.get(i);
     String[] tmpLemmas = lemmatizer.lemmatize(
                     st.toArray(new String[0]),
                     tmpPos.toArray((new String[0])));

     lemmas.add(Arrays.asList(tmpLemmas));
   }
} catch (IOException e) {
   e.printStackTrace();
}
```