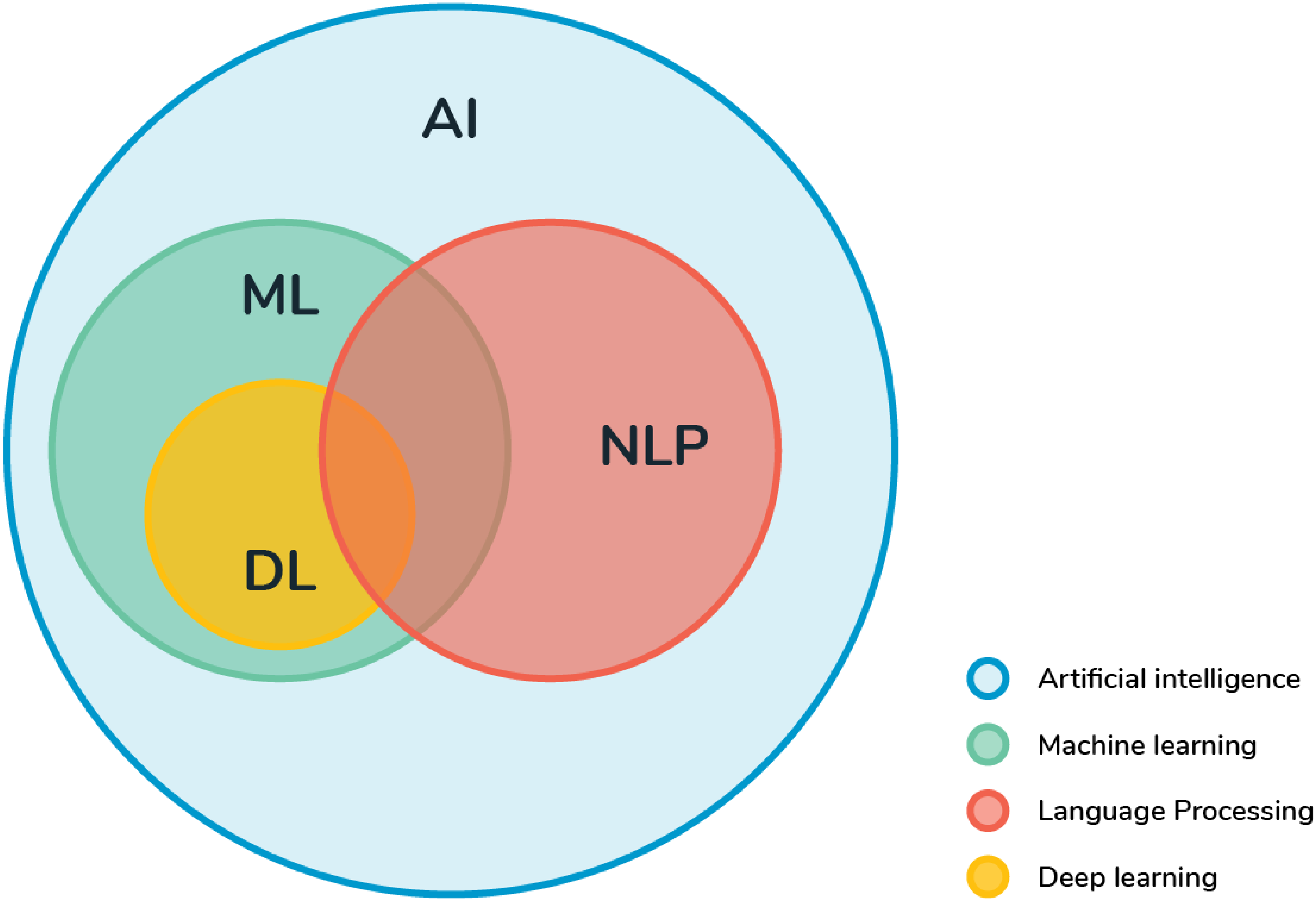
Preprocessing(1)


AI
ML
NLP
DL
Artificial intelligence
Machine learning
Language Processing
Deep learning

KoNLPy, NLTK

KoNLPy

KoNLPy 한국어 정보처리를 위한 파이썬 패키지

Corpus, Morpheme Analyzer, POS Tagging, …

http://konlpy.org/ko/latest/

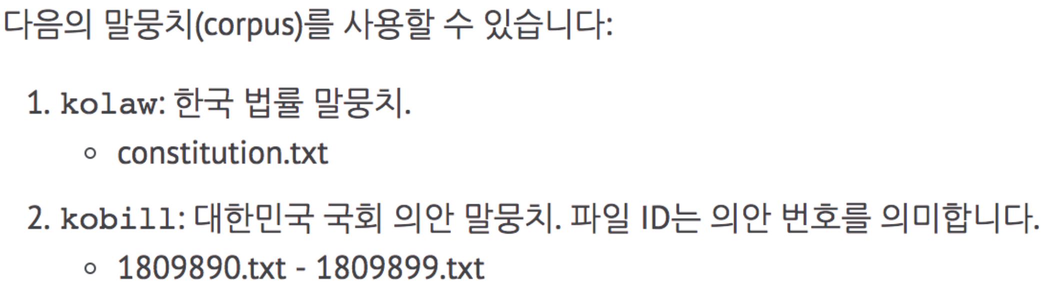
JDK 1.8 이상

JPype1 0.5.7 이상

Acknowledgement

박은정, 조성준, "KoNLPy: 쉽고 간결한 한국어 정보처리 파이썬 패키지",
제 26회 한글 및 한국어 정보처리 학술대회 논문집, 2014

Corpus



다음의 말뭉치(corpus)를 사용할 수 있습니다:

1. `kolaw`: 한국 법률 말뭉치.
   - constitution.txt
2. `kobill`: 대한민국 국회 의안 말뭉치. 파일 ID는 의안 번호를 의미합니다.
   - 1809890.txt - 1809899.txt

```
!pip install konlpy
```

```
from konlpy.corpus import kolaw

kolaw.fieids()

corpus = kolaw.open('constitution.txt').read()

print(len(corpus.split()))
print(corpus.splitlines()[:3])
```

NLTK

NLTK Building Python programs to work with human language data

Provides easy-to-use interfaces to over 50 corpora and lexical resources

classification, tokenization, stemming, tagging, parsing, and semantic reasoning

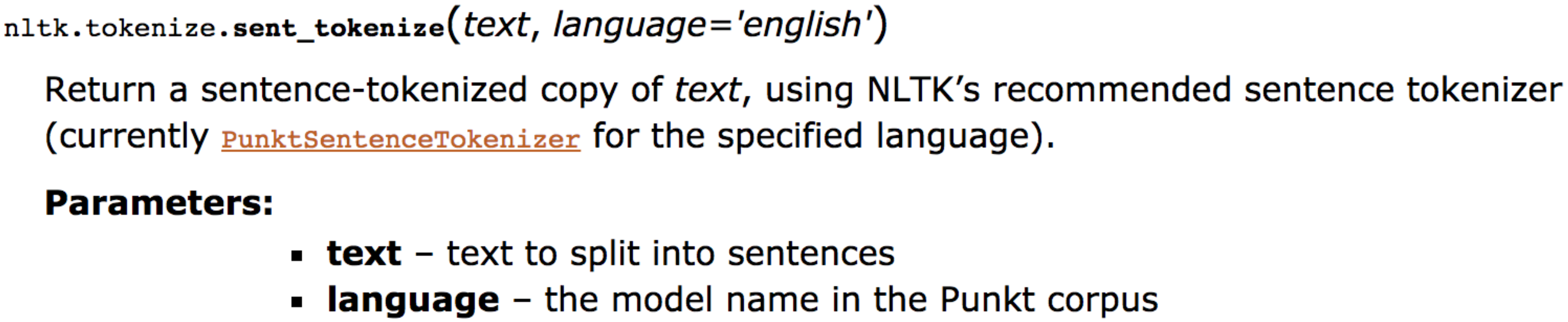
Corpus

```
!pip install nltk
```

```
import nltk

nltk.download('brown')
nltk.download('gutenberg')
```

```
from nltk.corpus import brown, gutenberg

corpus = brown.open('ca01').read()
print(len(corpus.split()))
print(corpus.splitlines()[:3])

corpus = gutenberg.open('austen-emma').read()
print(len(corpus.split()))
print(corpus.splitlines()[:3])
```

Tokenizing



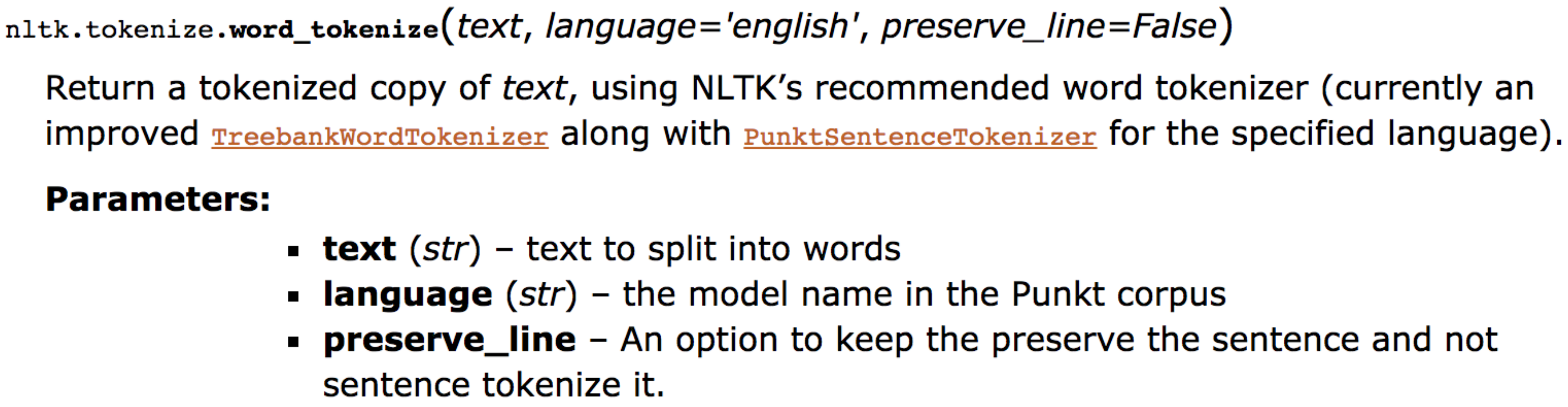
`nltk.tokenize.sent_tokenize`(*text*, *language='english'*)

Return a sentence-tokenized copy of *text*, using NLTK's recommended sentence tokenizer (currently `PunktSentenceTokenizer` for the specified language).

Parameters:

- **text** – text to split into sentences
- **language** – the model name in the Punkt corpus

```
from nltk.tokenize import sent_tokenize

nltk.download('punkt')

sentences = sent_tokenize(corpus)
print(len(sentences.split()))
print(sentences [:3])
```



nltk.tokenize.**word_tokenize**(*text*, *language='english'*, *preserve_line=False*)

Return a tokenized copy of *text*, using NLTK's recommended word tokenizer (currently an improved TreebankWordTokenizer along with PunktSentenceTokenizer for the specified language).

Parameters:

- **text** (*str*) – text to split into words
- **language** (*str*) – the model name in the Punkt corpus
- **preserve_line** – An option to keep the preserve the sentence and not sentence tokenize it.

```python
from nltk.tokenize import word_tokenize

words = word_tokenize(sentences[0])
print(len(words))
```