Slides on SpaCy

Gerhard Wohlgenannt [1] gwohlg@corp.ifmo.ru

[1] ITMO University, St. Petersburg, Russia

NLP4IS
February 24, 2020

Outline

Installation

https://spacy.io/

```
bash$ pip install –U spacy

bash$ spacy download en
```

```
# in virtual env
venv .env
source .env/bin/activate
pip install spacy

# possible with cuda / gpu
```

First Steps

```
import spacy
nlp = spacy.load('en')

# analyse a document with the model
doc = nlp(u'This␣is␣a␣sentence.')
```

First Steps (2)

```
text = open('war_and_peace.txt').read()
doc = nlp(text)

# Find named entities, phrases and concepts
for entity in doc.ents:
    print(entity.text, entity.label_)

# Determine semantic similarities
doc1 = nlp(u'the␣fries␣were␣gross')
doc2 = nlp(u'worst␣fries␣ever')
doc1.similarity(doc2)

# Hook in your own deep learning models
nlp.add_pipe(load_my_model(), before='parser')
```

First Steps (3)

```
from spacy import displacy

doc_ent = nlp(u'When␣Sebastian␣Thrun␣started␣working␣o
              u'in␣2007,␣few␣people␣outside␣of␣the␣com
displacy.serve(doc_ent, style='ent')
```

First Steps (4)

```
doc = nlp(u"Apple␣and␣banana␣are␣similar.␣Pasta␣and␣hi
apple = doc[0]
banana = doc[2]
pasta = doc[6]
hippo = doc[8]

assert apple.similarity(banana) > pasta.similarity(hip
assert apple.has_vector, banana.has_vector, pasta.has_
```

Spacy 101

https://spacy.io/usage/spacy-101

Features of Spacy

- Non-destructive tokenization
- POS, NER, Labelled dependency parsing, Syntax-driven sentence segmentation
- Support for 25+ languages
- 13 statistical models for 8 languages
- Pre-trained word vectors
- Easy deep learning integration, Export to numpy data arrays
- Built in visualizers for syntax and NER
- Convenient string-to-hash mapping
- **State-of-the-art speed. Robust, rigorously evaluated accuracy**