Natural Language Processing

Colin Jemmott
DSC 96, Fall 2018

Structured and Unstructured Data

Structured

- In a database
- Sorted and labeled with regular structure
- Proper types

Unstructured

- Just a bunch of stuff on the computer!
- Irregular and had ambiguities
- Difficult to understand using traditional programs

List examples of each.

Hutzler 571 Banana Slicer by Hutzler Manufacturing Co.

"What can I say about the 571B Banana Slicer that hasn't already been said about the wheel, penicillin, or the iPhone?"
Mrs Toledo

"Gone are the days of biting off slice-sized chunks of banana and spitting them onto a serving tray…. Next on my wish list: a kitchen tool for dividing frozen water into cube-sized chunks."
N. Krumpe

"As shown in the picture, the slices is curved from left to right. All of my bananas are bent the other way."
J. Anderson

80-90% of data is unstructured, and much of it is text. What can we do with it?

Syntax

Word segmentation

- This might be easy - or it “isn’t.”

Lemmatization and Stemming

- Reducing the inflectional forms of each word into a common base or root

Part-of-speech tagging

- Example: noun ("the book on the table") or verb ("to book a flight");

Semantics

Named entity recognition (NER)

- Which items in text map to proper names? What type (e.g. person, location)?

Machine translation

Sentiment Analysis

Natural language understanding, Question answering, Relationship extraction, Topic segmentation and recognition, Word sense disambiguation

NLTK

Tokenize and tag some text:

```
>>> import nltk
>>> sentence = """At eight o'clock on Thursday morning
... Arthur didn't feel very good."""
>>> tokens = nltk.word_tokenize(sentence)
>>> tokens
['At', 'eight', "o'clock", 'on', 'Thursday', 'morning',
'Arthur', 'did', "n't", 'feel', 'very', 'good', '.']
>>> tagged = nltk.pos_tag(tokens)
>>> tagged[0:6]
[('At', 'IN'), ('eight', 'CD'), ("o'clock", 'JJ'), ('on', 'IN'),
('Thursday', 'NNP'), ('morning', 'NN')]
```

NLTK

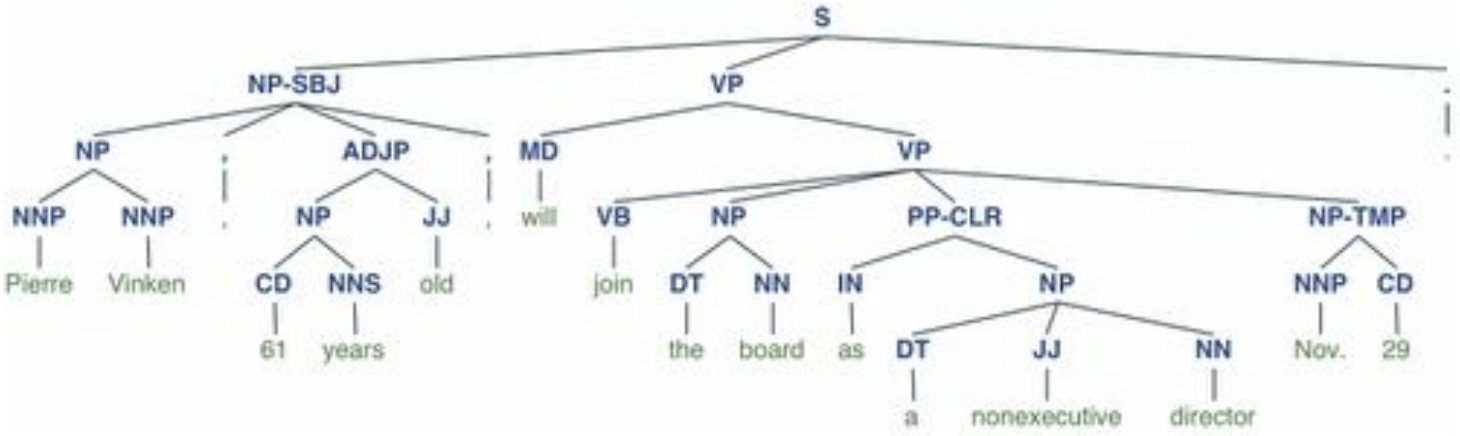
Identify named entities:

```
>>> entities = nltk.chunk.ne_chunk(tagged)
>>> entities
Tree('S', [('At', 'IN'), ('eight', 'CD'), ("o'clock", 'JJ'),
           ('on', 'IN'), ('Thursday', 'NNP'), ('morning', 'NN'),
       Tree('PERSON', [('Arthur', 'NNP')]),
           ('did', 'VBD'), ("n't", 'RB'), ('feel', 'VB'),
           ('very', 'RB'), ('good', 'JJ'), ('.', '.')])
```

NLTK

Display a parse tree:

```
>>> from nltk.corpus import treebank
>>> t = treebank.parsed_sents('wsj_0001.mrg')[0]
>>> t.draw()
```



Other NLP Tools

Commercial solutions (Google, Microsoft, Amazon, IBM, etc)

- Translation: don’t DIY

SpaCy

- Similar performance to NLTK
- Many fewer options
- ~500x faster