ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DIPARTIMENTO DI
SCIENZE STATISTICHE "PAOLO FORTUNATI"

9 - Information Extraction

Giovanni Della Lunga

giovanni.dellalunga@unibo.it

Halloween Conference in Quantitative Finance

Bologna - October 26-28, 2021

Subsection 1

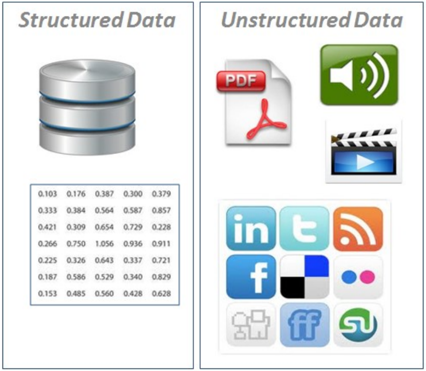
Unstructured Data Analysis

Unstructured Data Analysis

- Data generated from conversations, declarations or even tweets are examples of unstructured data.
- Unstructured data doesn't fit neatly into the traditional row and column structure of relational databases, and represent the vast majority of data available in the actual world.
- It is messy and hard to manipulate.
- Nevertheless, thanks to the advances in disciplines like machine learning a big revolution is going on regarding this topic.

Introduction

- In order to act on unstructured form of information (data), the ML models have to perform one of the crucial processes called Information Extraction(IE).
- Information Extraction is the process of retrieving key information intertwined within the unstructured data.
- In other words, extracting structured data from the unstructured data.



0.103	0.176	0.387	0.300	0.379
0.333	0.384	0.564	0.587	0.857
0.421	0.309	0.654	0.729	0.228
0.266	0.750	1.056	0.936	0.911
0.225	0.326	0.643	0.337	0.721
0.187	0.586	0.529	0.340	0.829
0.153	0.485	0.560	0.428	0.628

Introduction

The goal of this section is to answer the following questions:

- How can we build a system that extracts structured data, such as tables, from unstructured text?
- What are some robust methods for identifying the entities and relationships described in a text?
- Which corpora are appropriate for this work, and how do we use them for training and evaluating our models?

Along the way, we'll apply techniques from the previous sections to the problems of chunking and named-entity recognition.

Introduction

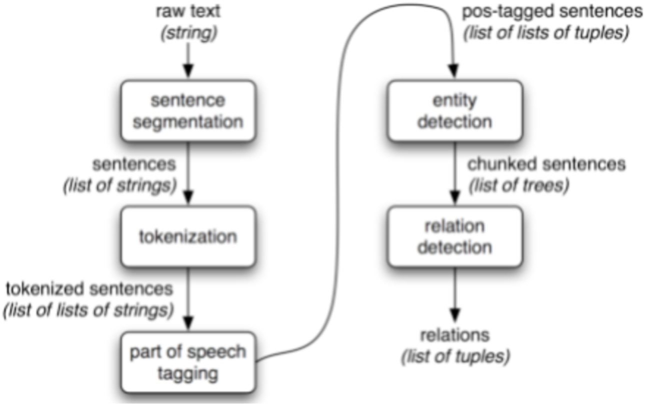
- One approach to this problem involves building a very general representation of meaning.
- In order to simplify the problem at hand, we will take a different approach, deciding in advance that we will only look for very specific kinds of information in text, such as the relation between organizations and locations;
- Rather than trying to use text like to answer a question directly, we first **convert the unstructured data** of natural language sentences into a **structured data**.
- Then we apply the benefits of powerful query tools such as SQL.
- This method of getting meaning from text is called **Information Extraction**.

Introduction

- In the following slide we shows the architecture for a simple information extraction system based on the NLTK project (see references).
- It begins by processing a document using several of the procedures already discussed: first, the raw text of the document is split into sentences using a sentence segmenter, and each sentence is further subdivided into words using a tokenizer.
- Next, each sentence is tagged with part-of-speech tags, which will prove very helpful in the next step, named entity detection.
- In this step, we search for mentions of potentially interesting entities in each sentence.
- Finally, we use relation detection to search for likely relations between different entities in the text.

Information Extraction Architecture

source: *Bird S. et al. Natural Language Processing with Python, Chapter 7*



Subsection 2

Named Entity Recognition

Named Entity Recognition

- A crucial component in IE systems is Named Entity Recognition (NER).
- Named-entity recognition is the problem of identifying and classifying entities into categories such as the names of people, locations, organizations, the expressions of quantities, times, measurements, monetary values, and so on.
- In general terms, entities refer to names of people, organizations (e.g. United Nations, American Airlines), places/cities (Rome, Boston), etc.

Named Entity Recognition

- *The fourth Wells account moving to another agency is the packaged paper-products division of Georgia-Pacific Corp., which arrived at Wells only last fall. Like Hertz and the History Channel, it is also leaving for an Omnicom-owned agency, the BBDO South unit of BBDO Worldwide. BBDO South in Atlanta, which handles corporate advertising for Georgia-Pacific, will assume additional duties for brands like Angel Soft toilet tissue and Sparkle paper towels, said Ken Haldin, a spokesman for Georgia-Pacific in Atlanta.*
- The question is: **which companies are based in Atlanta?**

Taken from Bird S. et al. *Natural Language Processing with Python* O'Reilly (2009)

Extracting Information from Text

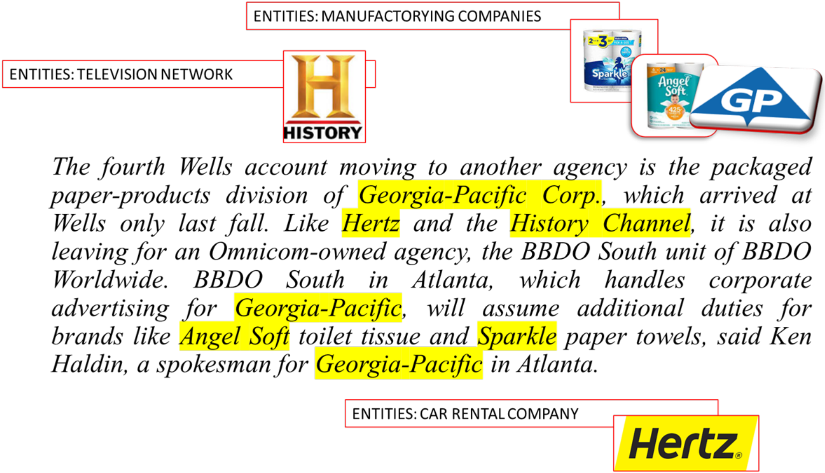
ENTITIES: ADVERTISING AGENCY

Yoo-Hoo
A new chocolate energy drink
A taste of Europe flying in the U.S.A.
TWA

The fourth Wells account moving to another agency is the packaged paper-products division of Georgia-Pacific Corp., which arrived at Wells only last fall. Like Hertz and the History Channel, it is also leaving for an Omnicom-owned agency, the BBDO South unit of BBDO Worldwide. BBDO South in Atlanta, which handles corporate advertising for Georgia-Pacific, will assume additional duties for brands like Angel Soft toilet tissue and Sparkle paper towels, said Ken Haldin, a spokesman for Georgia-Pacific in Atlanta.

Wells Rich Greene BDDP

Extracting Information from Text



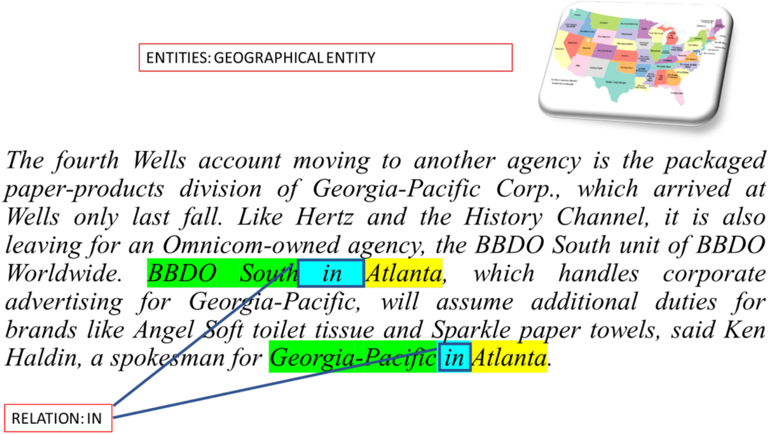
ENTITIES: MANUFACTORYING COMPANIES

ENTITIES: TELEVISION NETWORK

The fourth Wells account moving to another agency is the packaged paper-products division of Georgia-Pacific Corp., which arrived at Wells only last fall. Like Hertz and the History Channel, it is also leaving for an Omnicom-owned agency, the BBDO South unit of BBDO Worldwide. BBDO South in Atlanta, which handles corporate advertising for Georgia-Pacific, will assume additional duties for brands like Angel Soft toilet tissue and Sparkle paper towels, said Ken Haldin, a spokesman for Georgia-Pacific in Atlanta.

ENTITIES: CAR RENTAL COMPANY

Extracting Information from Text



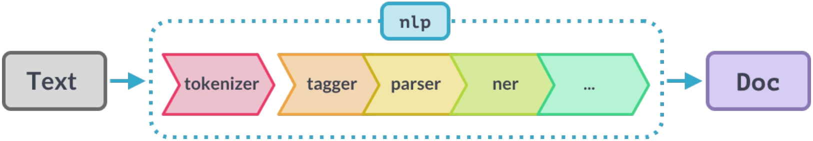
ENTITIES: GEOGRAPHICAL ENTITY

The fourth Wells account moving to another agency is the packaged paper-products division of Georgia-Pacific Corp., which arrived at Wells only last fall. Like Hertz and the History Channel, it is also leaving for an Omnicom-owned agency, the BBDO South unit of BBDO Worldwide. BBDO South in Atlanta, which handles corporate advertising for Georgia-Pacific, will assume additional duties for brands like Angel Soft toilet tissue and Sparkle paper towels, said Ken Haldin, a spokesman for Georgia-Pacific in Atlanta.

RELATION: IN

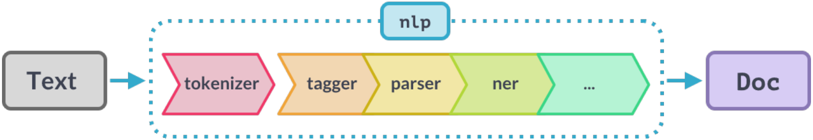
Extracting Information from Text: Using spaCy

- Download and install a trained pipeline (in this case ‘en_core_web_sm‘), you can load it via spacy.load.
- This will return a Language object containing all components and data needed to process text. We call it ‘spacy_nlp‘. Calling the nlp object on a string of text will return a processed Doc.
- In particular, When you call ‘spacy_nlp‘ on a text, spaCy first tokenizes the text to produce a Doc object.

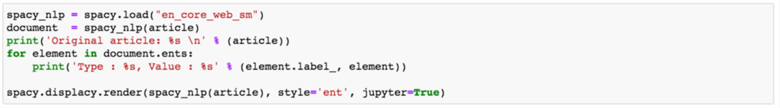


Extracting Information from Text: Using spaCy

- The Doc is then processed in several different steps – this is also referred to as the processing pipeline.
- The pipeline used by the trained pipelines typically include a tagger, a lemmatizer, a parser and an entity recognizer. Each pipeline component returns the processed Doc, which is then passed on to the next component.

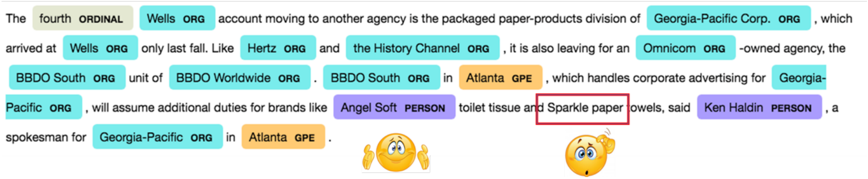


Extracting Information from Text: Using spaCy



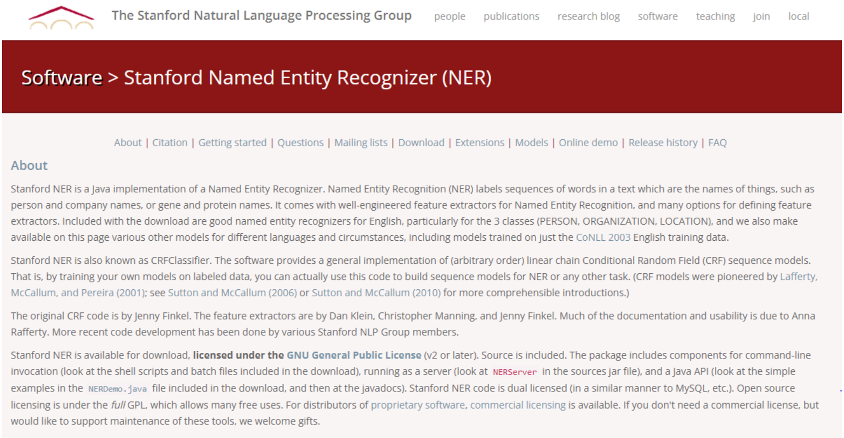
```
spacy_nlp = spacy.load("en_core_web_sm")
document  = spacy_nlp(article)
print('Original article: %s \n' % (article))
for element in document.ents:
    print('Type : %s, Value : %s' % (element.label_, element))

spacy.displacy.render(spacy_nlp(article), style='ent', jupyter=True)
```



The fourth ORDINAL Wells ORG account moving to another agency is the packaged paper-products division of Georgia-Pacific Corp. ORG , which arrived at Wells ORG only last fall. Like Hertz ORG and the History Channel ORG , it is also leaving for an Omnicom ORG -owned agency, the BBDO South ORG unit of BBDO Worldwide ORG . BBDO South ORG in Atlanta GPE , which handles corporate advertising for Georgia-Pacific ORG , will assume additional duties for brands like Angel Soft PERSON toilet tissue and Sparkle paper towels, said Ken Haldin PERSON , a spokesman for Georgia-Pacific ORG in Atlanta GPE .

Extracting Information from Text: Using NLTK with Stanford NER



The Stanford Natural Language Processing Group
people publications research blog software teaching join local

Software > Stanford Named Entity Recognizer (NER)

About | Citation | Getting started | Questions | Mailing lists | Download | Extensions | Models | Online demo | Release history | FAQ

About

Stanford NER is a Java implementation of a Named Entity Recognizer. Named Entity Recognition (NER) labels sequences of words in a text which are the names of things, such as person and company names, or gene and protein names. It comes with well-engineered feature extractors for Named Entity Recognition, and many options for defining feature extractors. Included with the download are good named entity recognizers for English, particularly for the 3 classes (PERSON, ORGANIZATION, LOCATION), and we also make available on this page various other models for different languages and circumstances, including models trained on just the CoNLL 2003 English training data.

Stanford NER is also known as CRFClassifier. The software provides a general implementation of (arbitrary order) linear chain Conditional Random Field (CRF) sequence models. That is, by training your own models on labeled data, you can actually use this code to build sequence models for NER or any other task. (CRF models were pioneered by Lafferty, McCallum, and Pereira (2001); see Sutton and McCallum (2006) or Sutton and McCallum (2010) for more comprehensible introductions.)

The original CRF code is by Jenny Finkel. The feature extractors are by Dan Klein, Christopher Manning, and Jenny Finkel. Much of the documentation and usability is due to Anna Rafferty. More recent code development has been done by various Stanford NLP Group members.

Stanford NER is available for download, **licensed under the GNU General Public License** (v2 or later). Source is included. The package includes components for command-line invocation (look at the shell scripts and batch files included in the download), running as a server (look at `NERServer` in the sources jar file), and a Java API (look at the simple examples in the `NERDemo.java` file included in the download, and then at the javadocs). Stanford NER code is dual licensed (in a similar manner to MySQL, etc.). Open source licensing is under the *full* GPL, which allows many free uses. For distributors of proprietary software, commercial licensing is available. If you don't need a commercial license, but would like to support maintenance of these tools, we welcome gifts.

Example : Reading the Newspaper

- Notebook: 09-information-extraction
- Libraries: NLTK, Stanford Group NER
- https://nlp.stanford.edu/software/

Comparing Results

- Natural language processing applications are characterized by complex interdependent decisions that require large amounts of prior knowledge.
- In this case, as you can see, the system designed by Stanford did not achieve the same result as spaCy, but it is pure accident; in fact, a lot depends on how well the models have been trained and with how much data.
- For this reason, in case there is a need to perform a task like this, the best thing to do is to use multiple tools and compare the results, in order to find the best one in terms of performance and response.

Subsection 3

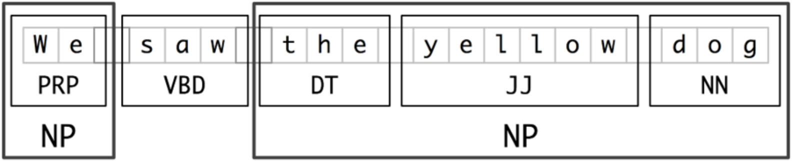
Chunking

What is chunking?

- Chunking is a process of extracting phrases from unstructured text, which means analyzing a sentence to identify the constituents(Noun Groups, Verbs, verb groups, etc.)
- It works on top of POS tagging.
- It uses POS-tags as input and provides chunks as output.
- Chunking is one of the rule-based text extraction processes which is used for building Named-Entity recognition models.

What is chunking?

- In chucking, a chunker chunks the phrases that are meaningful in a text.



source: *Bird S. et al. "Natural Language Processing with Python" O'Reilly (2009) Ch. 7*

Why chunking is important?

- Simply breaking text into words in many situations isn't very helpful.
- It's very crucial to know that sentence involves a person, a date, places, etc.. (different entities);
- Chunking can break sentences into phrases that are more useful than individual words and yield meaningful results.
- Chunking is very important when you want to extract information from text such as locations, person names (entity extraction).

Chunking

Let's understand it from scratch. A sentence typically follows a hierarchical structure consisting of the following components.

sentence → **clauses** → **phrases** → **words**

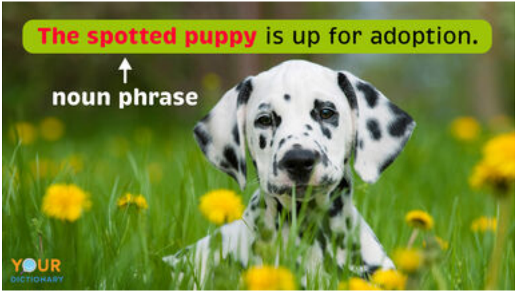
Group of words make up phrases and there are five major categories.

- Noun Phrase (NP)
- Verb phrase (VP)
- Adjective phrase (ADJP)
- Adverb phrase (ADVP)
- Prepositional phrase (PP)

Chunking

Noun Phrases

- Noun phrases are groups of words that function like nouns. Typically, they act as subjects, objects or prepositional objects in a sentence.
- Noun phrases are simply nouns with modifiers. Just as nouns can act as subjects, objects and prepositional objects, so can noun phrases.



Chunking

Noun Phrases

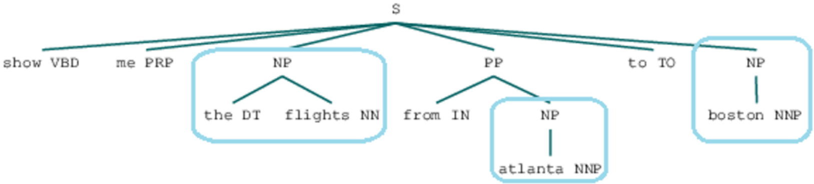
- Noun phrases are groups of words that function like nouns. Typically, they act as subjects, objects or prepositional objects in a sentence.
- Noun phrases are simply nouns with modifiers. Just as nouns can act as subjects, objects and prepositional objects, so can noun phrases.

$$< DT >? < JJ > * < NN >.$$

THE	- Determinative
SPOTTED	- Adjective
PUPPY	- Noun

Regex Based Chunking

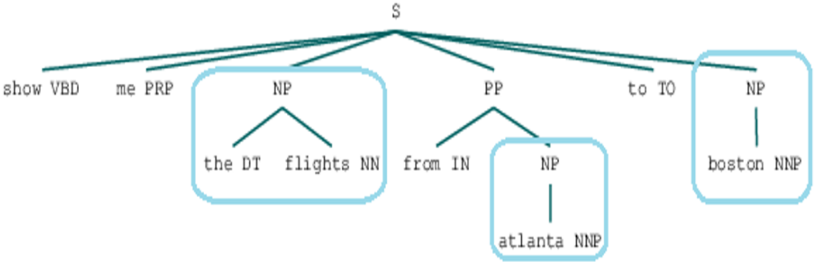
- The chunker is built upon a set of production rules, otherwise known as Grammar Rules.
- For instance, in the case of NER, grammar can be a pattern to match a Noun Phrase, since Named-entities are mostly nouns.



source: *https://payodatechnologyinc.medium.com/extract-meaningful-information-from-big-data-using-nlp-and-machine-learning*

Regex Based Chunking

As you can see chunker rules are based on PoS tags, Pos tagging becomes the necessary task before chunking.



source: *https://payodatechnologyinc.medium.com/extract-meaningful-information-from-big-data-using-nlp-and-machine-learning*

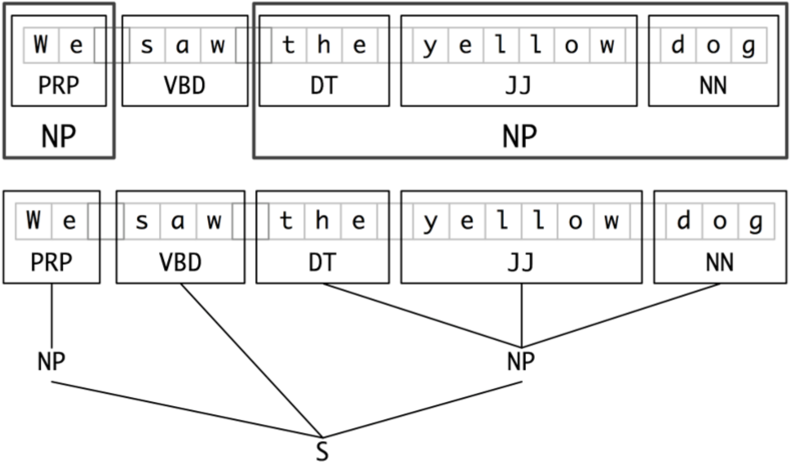
Regex Based Chunking

Now lets under understand this concept with python experiment. We are going to introduce a grammar in which NP (Noun phrase) is combined by

- DT? → one or zero determiner
- JJ* → zero or more adjectives
- NN → Noun

and we parse this grammar by NLTK defined regular expression parser. As we will see, whole sentence **S** is divided into chunks and represented in tree-like structures. Based on defined grammar, an internally tree-like structure is created. So you can define your grammar, based on that sentence will be chunked.

Regex Based Chunking



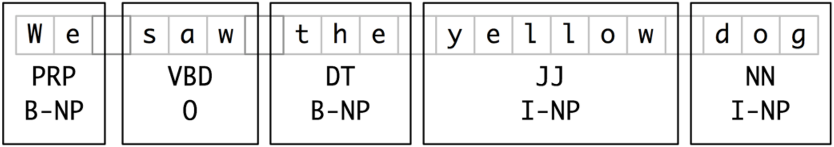
Example : Regex Based Chunking

- Notebook: 09-information-extraction
- Libraries: NLTK, Stanford Group NER

Training tagger based chunker

The IOB tagging scheme

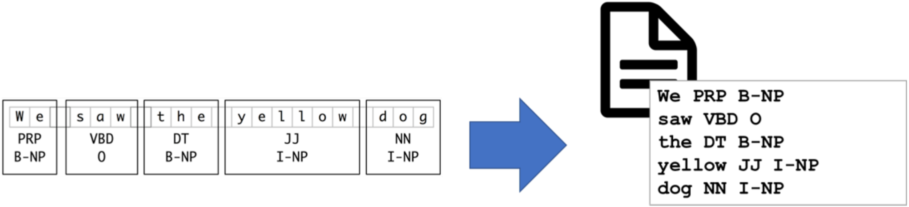
- In this scheme, each token is tagged with one of three special chunk tags, I (inside), O (outside), or B (begin).
- A token is tagged as B if it marks the beginning of a chunk. Subsequent tokens within the chunk are tagged I. All other tokens are tagged O.
- The B and I tags are suffixed with the chunk type, e.g. B-NP, I-NP.
- Of course, it is not necessary to specify a chunk type for tokens that appear outside a chunk, so these are just labeled O.



We	saw	the	yellow	dog
PRP	VBD	DT	JJ	NN
B-NP	O	B-NP	I-NP	I-NP

Training tagger based chunker

- We'll use the ‘conll2000‘ corpus from NLTK package for training chunker.
- The CoNLL 2000 corpus contains 270k words of Wall Street Journal text, divided into "train" and "test" portions, annotated with part-of-speech tags and chunk tags in the IOB format. We can access the data using nltk.corpus.conll2000.
- It specifies where the chunk begins and ends, along with its types.
- A POS tagger can be trained on these IOB tags



```
We PRP B-NP
saw VBD O
the DT B-NP
yellow JJ I-NP
dog NN I-NP
```

Relation Extraction

- Once named entities have been identified in a text, we then want to extract the relations that exist between them.
- We will typically be looking for relations between specified types of named entity.
- One way of approaching this task is to initially look for all triples of the form (X, α, Y), where X and Y are named entities of the required types, and α is the string of words that link X and Y.
- We can then use regular expressions to pull out just those instances of α that express the relation that we are looking for.

Example : Transactional Data

- Notebook: 09-information-extraction
- Libraries: NLTK, Stanford Group NER