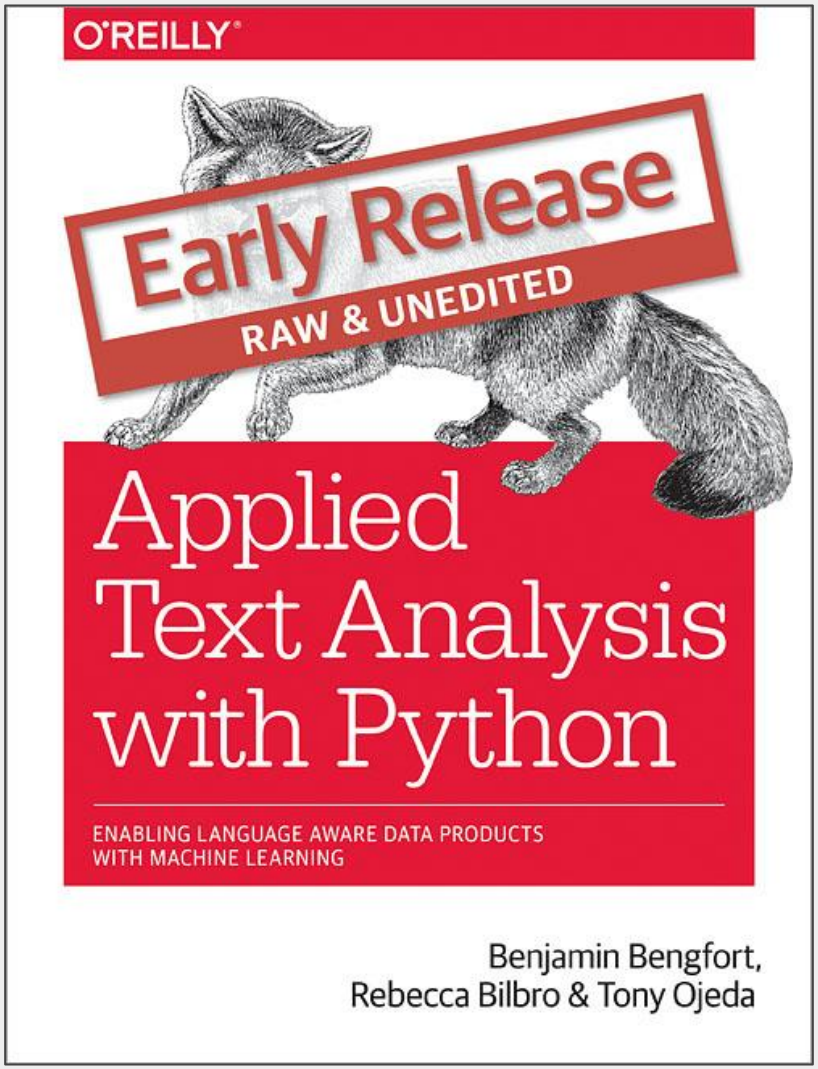
Natural Language Processing

CEB Day 6: February 10, 2017

DistrictDataLabs

Applied Text Analysis with Python

The discussion in these slides and in the notebooks is currently being expanded on!



Intro to Natural Language Processing

What is Language?

Or, why should we analyze it?

DistrictDataLabs

Crab

What does this mean?

/kræb/ **Crab**

Is the symbol representative of its meaning?

sound	symbol	sight
/kræb/	**Crab**	

Or is it just a mapping in a mental lexicon?

sound	symbol	sight
/kávouras/	**κάβουρας**	

These symbols are arbitrary

What is Language?

Linguistic symbols:

- Not acoustic “things”
- Not mental processes

Critical implications for:

- Literature
- Linguistics
- Computer Science
- Artificial Intelligence

What is Natural Language Processing?

The science that has been developed around the facts of language passed through three stages before finding its true and unique object. First something called "grammar" was studied. This study, initiated by the Greeks and continued mainly by the French, was based on logic. It lacked a scientific approach and was detached from language itself. Its only aim was to give rules for distinguishing between correct and incorrect forms; it was a normative discipline, far removed from actual observation, and its scope was limited.

-- Ferdinand de Saussure

What is Natural Language Processing?

The science that has been developed around the facts of language passed through three stages before finding its true and unique object. First something called "grammar" was studied. This study, initiated by the Greeks and continued mainly by the French, was based on logic. It lacked a scientific approach and was detached from language itself. Its only aim was to give rules for distinguishing between correct and incorrect forms; it was a normative discipline, far removed from actual observation, and its scope was limited.

-- Ferdinand de Saussure

Formal vs. Natural Languages

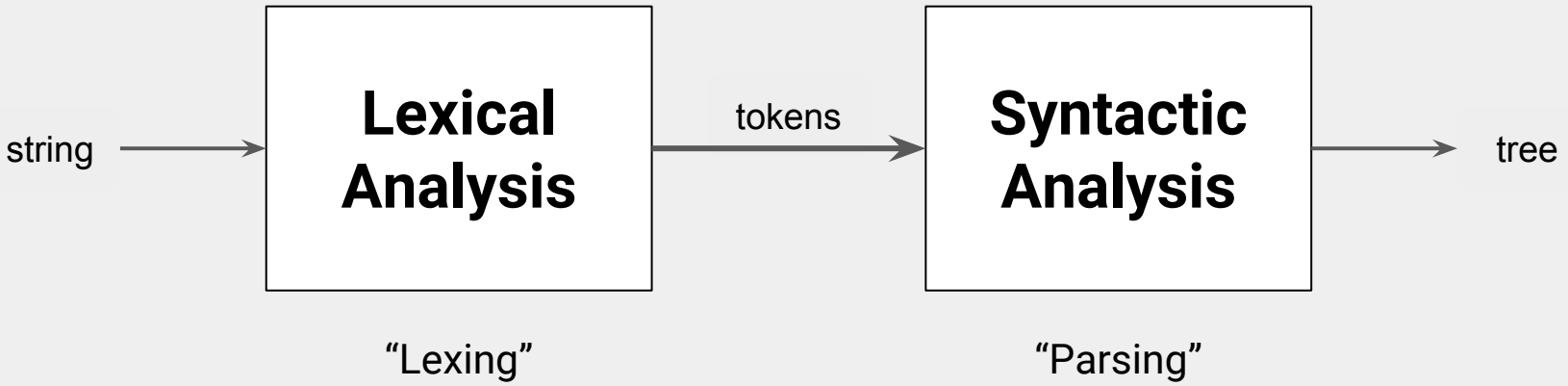
Formal Languages

- Strict, unchanging rules defined by grammars and parsed by regular expressions
- Generally application specific (chemistry, math)
- Literal: exactly what is said is meant.
- No ambiguity
- Parsable by regular expressions
- Inflexible: no new terms or meaning.

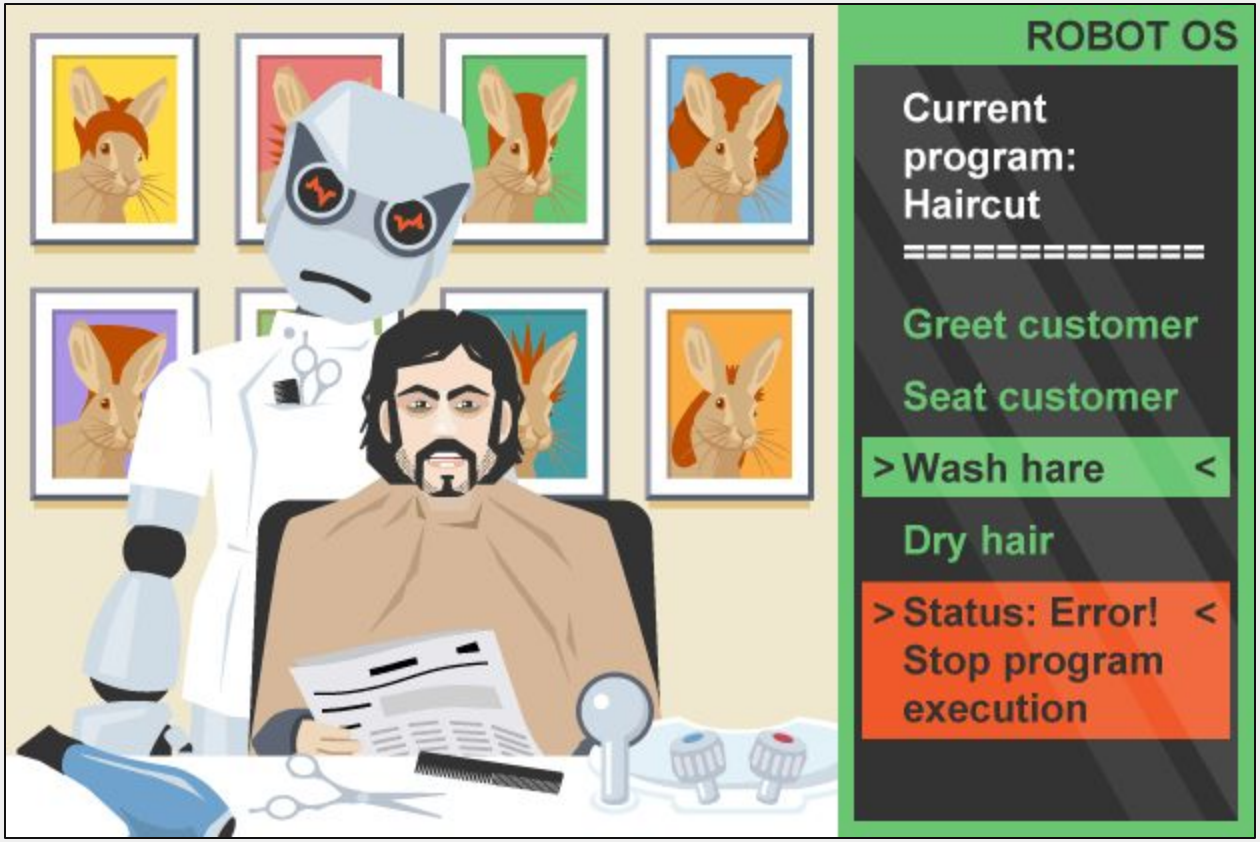
Natural Languages

- Flexible, evolving language that occurs naturally in human communication
- Unspecific and used in many domains and applications
- Redundant and verbose in order to make up for ambiguity
- Expressive
- Difficult to parse
- Very flexible even in narrow contexts

Computer science has traditionally focused on formal languages.



Who has written a compiler or interpreter?

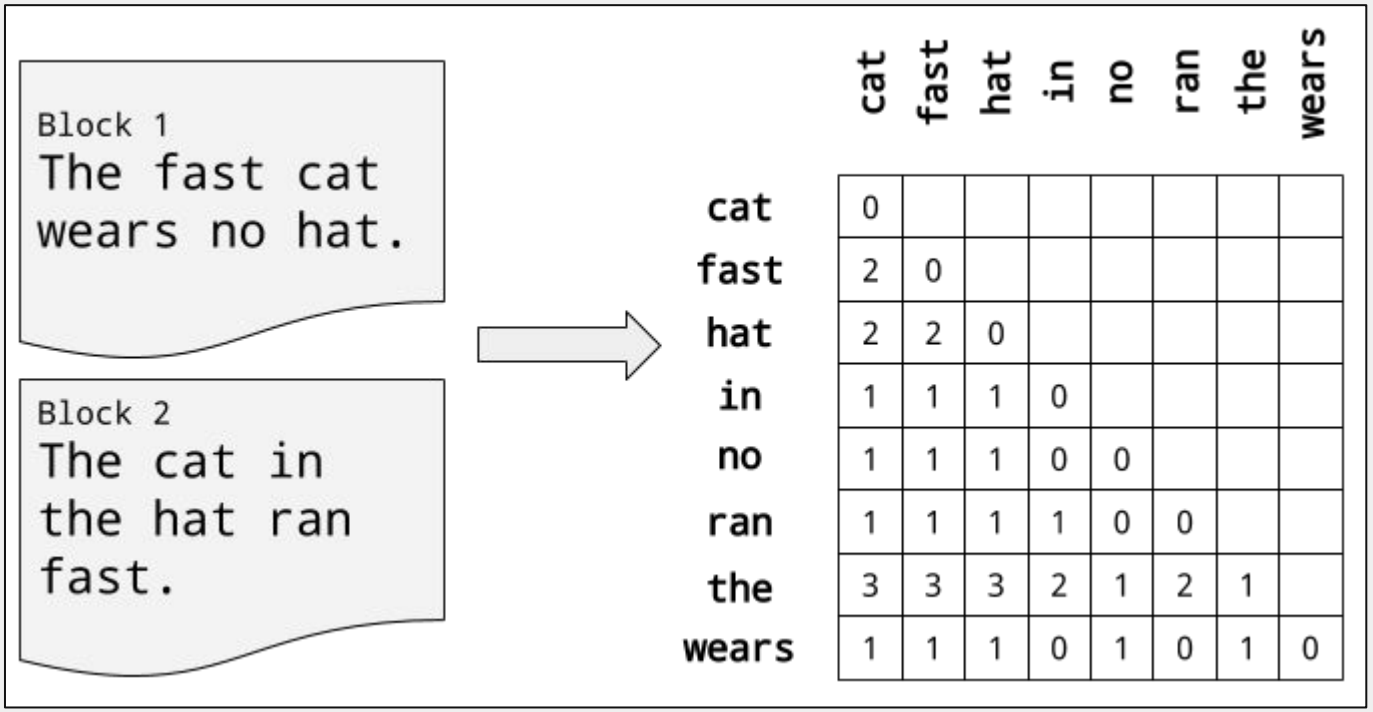


Syntax Errors are your friend! http://bbc.in/23pSPRq

However, ambiguity is required for understanding when communicating between people with diverse experience.

Most of the time. http://bit.ly/1XlOtHv

Natural Language Processing requires flexibility, which generally comes from machine learning.



	cat	fast	hat	in	no	ran	the	wears
cat	0							
fast	2	0						
hat	2	2	0					
in	1	1	1	0				
no	1	1	1	0	0			
ran	1	1	1	1	0	0		
the	3	3	3	2	1	2	1	
wears	1	1	1	0	1	0	1	0

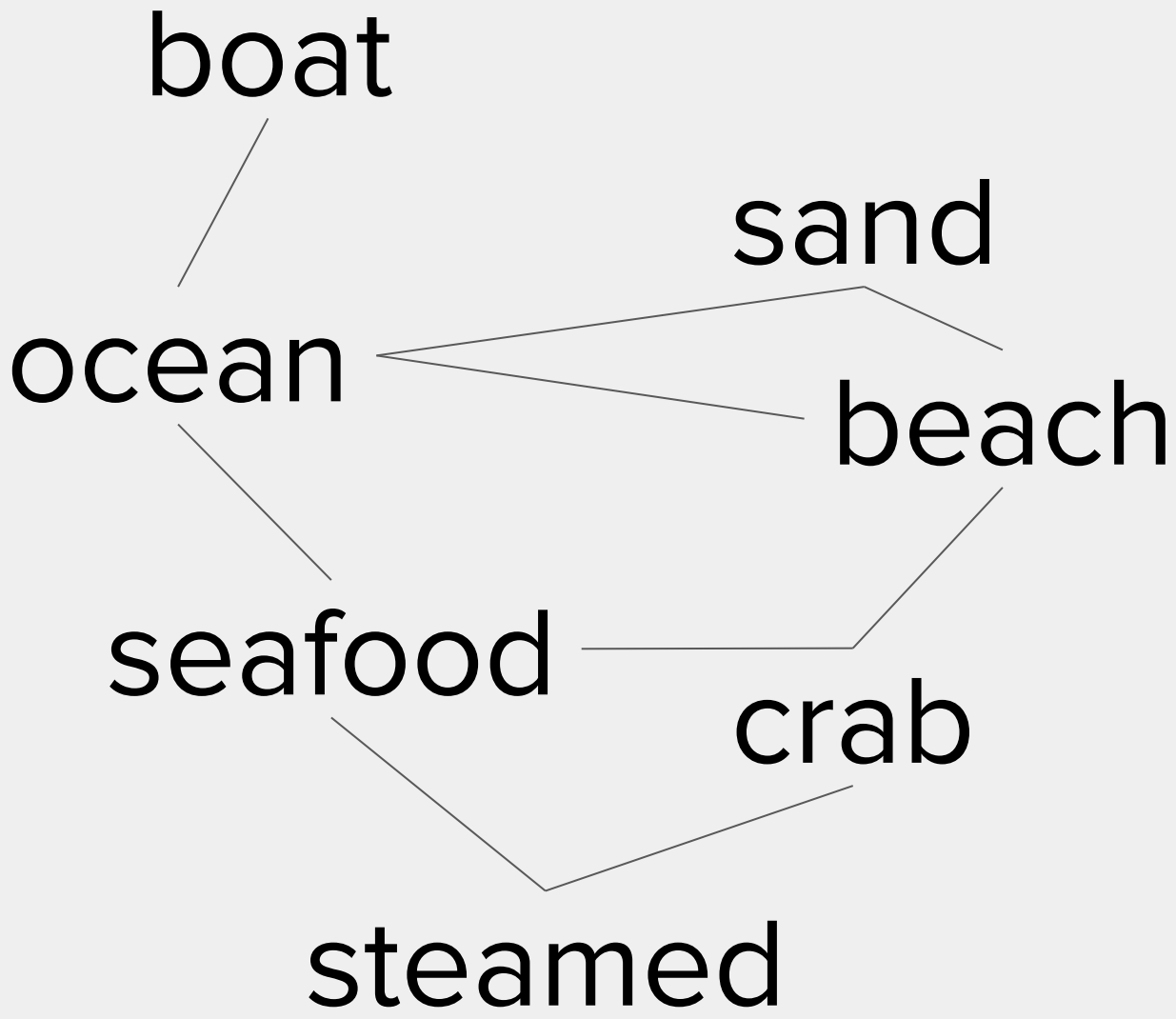
Language models are used for flexibility

Intuition: Language is Predictable (if flexible)

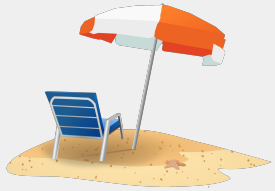
"There was a ton of traffic on the beltway so I was _____."

"At beach we watched the _____."

"Watch out for that _____!"



These models predict relationships between tokens.



But they can’t understand meaning (yet)

DistrictDataLabs

So please keep in mind:

Tokens != Words

- Substrings
- Only structural
- Data

```
“bearing”
“shouldn’t”
```

- Objects
- Contains a “sense”
- Meaning

```
to bear.verb-1
should.auxverb-3
not.adverb-1
```

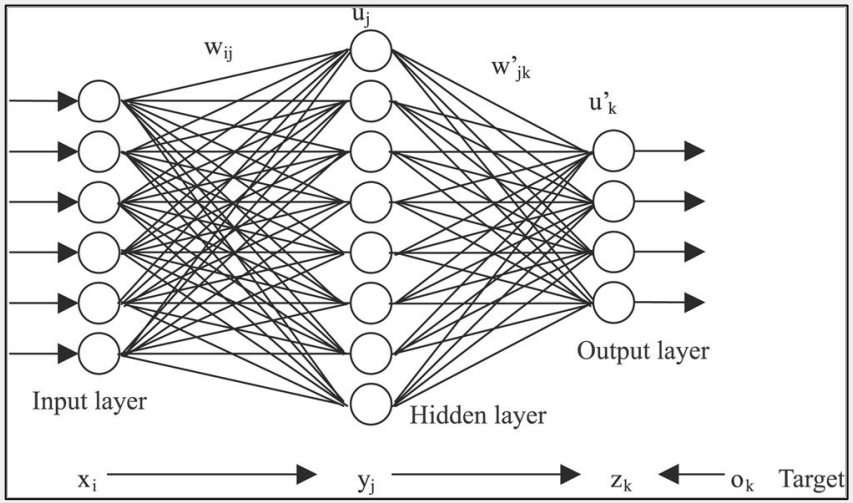
Connectionist vs Symbolic Models

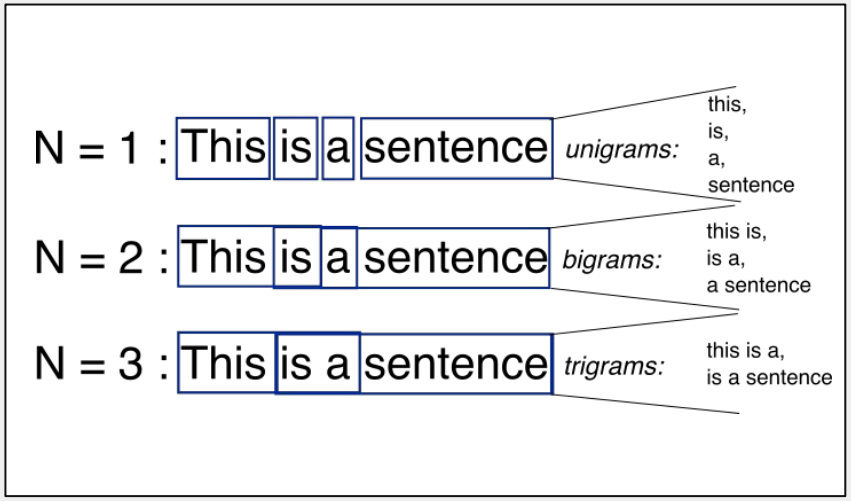
Language is modeled as connected units with emergent behavior. Context is non-human.

Model: Recurrent Tensor NN

Language is modeled as discrete chunks, context is embedded locally and human understandable.

Model: Kneser-Ney 5gram





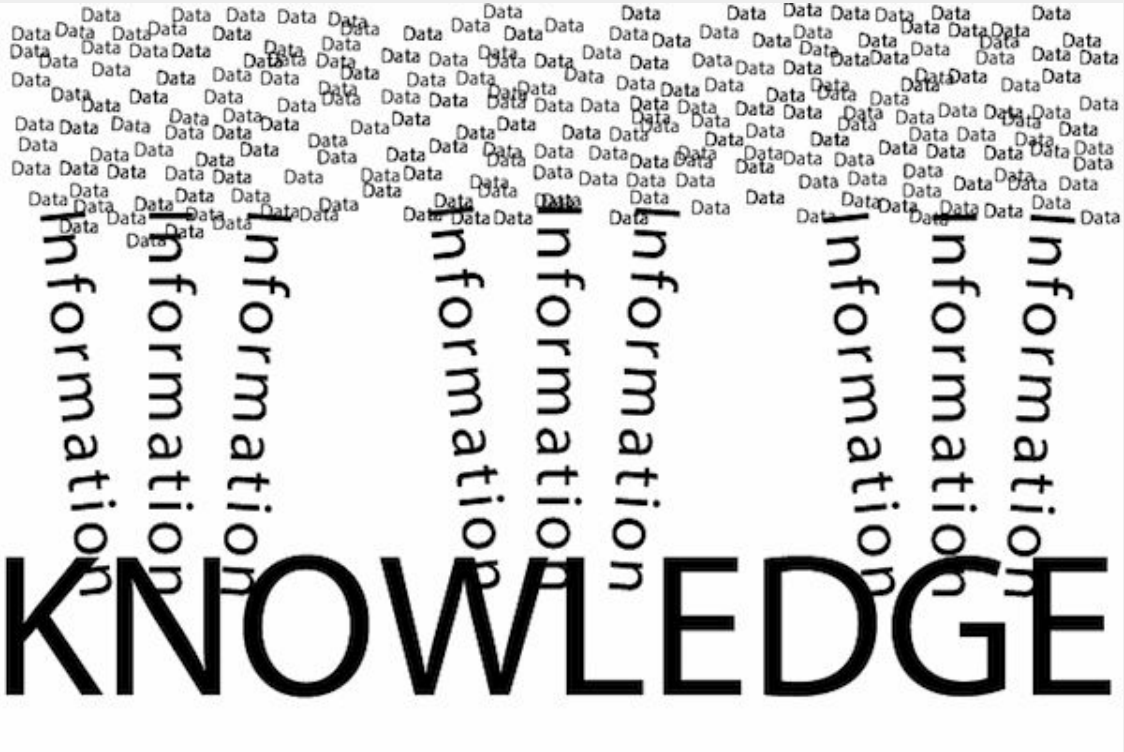
The State of the Art

- Academic design for use alongside intelligent agents (AI discipline)
- Relies on formal models or representations of knowledge & language
- Models are adapted and augmented through probabilistic methods and machine learning.
- A small number of algorithms comprise the standard framework.

Traditional NLP Applications

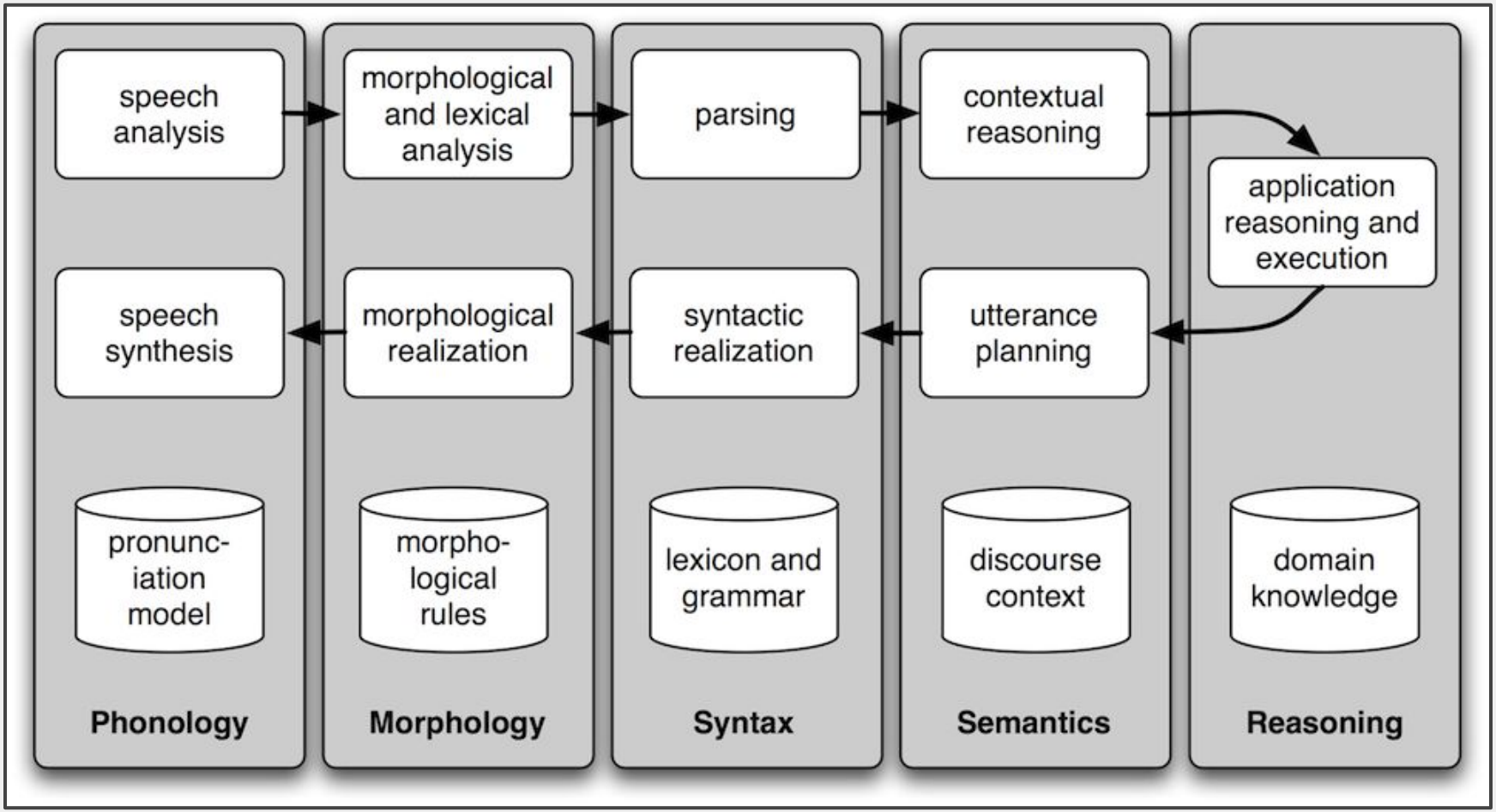
- Summarization
- Reference Resolution
- Machine Translation
- Language Generation
- Language Understanding
- Document Classification
- Author Identification
- Part of Speech Tagging
- Question Answering
- Information Extraction
- Information Retrieval
- Speech Recognition
- Sense Disambiguation
- Topic Recognition
- Relationship Detection
- Named Entity Recognition

What is Required?



Domain Knowledge

A Corpus in the Domain



The NLP Pipeline

Morphology

The study of the forms of things, words in particular.

Consider pluralization for English:

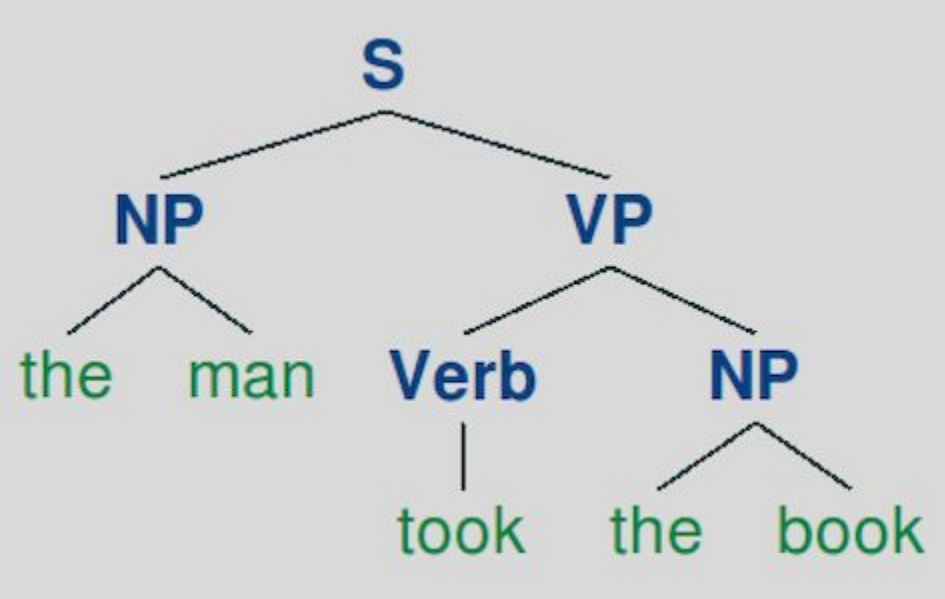
- Orthographic Rules: puppy → puppies
- Morphological Rules: goose → geese or fish

Major parsing tasks:

stemming, lemmatization and tokenization.

Syntax

The study of the rules for the formation of sentences.



Major tasks:

chunking, parsing, feature parsing, grammars

Semantics

The study of meaning.

- I see what I eat.
- I eat what I see.
- He poached salmon.

Major Tasks

Frame extraction, creation of TMRs

“The man hit the building with the baseball bat”

```
{
    "subject": {"text": "the man", "sense": human-agent},
    "predicate": {"text": "hit", "sense": strike-physical-force},
    "object": {"text": "the building", "sense": habitable-structure},
    "instrument": {"text": "with the baseball bat" "sense": sports-equipment}
}
```

Thematic Meaning Representation

Recent NLP Applications

- Yelp Insights
- Winning Jeopardy! IBM Watson
- Computer assisted medical coding (3M Health Information Systems)
- Geoparsing -- CLAVIN (built by Charlie Greenbacker)
- Author Identification (classification/clustering)
- Sentiment Analysis (RTNNs, classification)
- Language Detection
- Event Detection
- Google Knowledge Graph
- Named Entity Recognition and Classification
- Machine Translation
- Image + Language Processing

Applications are BIG data

- Examples are easier to create than rules.
- Rules and logic miss frequency and language dynamics
- More data is better for machine learning, relevance is in the long tail
- Knowledge engineering is not scalable
- Computational linguistics methodologies are stochastic

The Natural Language Toolkit (NLTK)

What is NLTK?

- Python interface to over 50 corpora and lexical resources
- Focus on Machine Learning with specific domain knowledge
- Free and Open Source
- Numpy and Scipy under the hood
- Fast and Formal

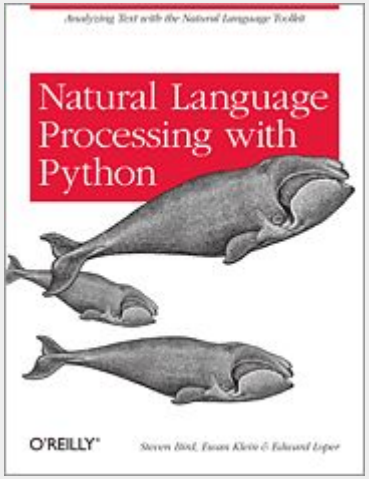
What is NLTK?

Suite of libraries for a variety of academic text processing tasks:

- tokenization, stemming, tagging,
- chunking, parsing, classification,
- language modeling, logical semantics

Pedagogical resources for teaching NLP theory in Python ...

Who Wrote NLTK?



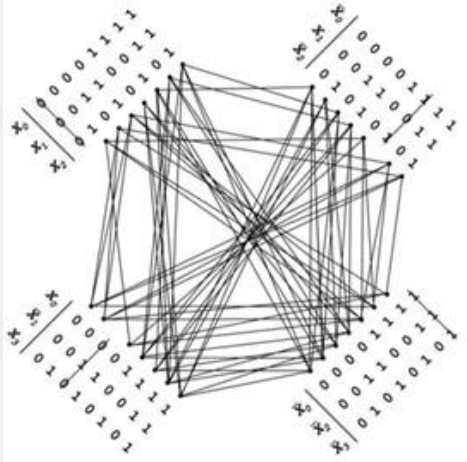
Steven Bird

Associate Professor
University of Melbourne
Senior Research Associate, LDC

Ewan Klein

Professor of Language Technology
University of Edinburgh.

Batteries Included



NLTK = Corpora + Algorithms
Ready for Research!

What is NLTK not?

- Production ready out of the box*
- Lightweight
- Generally applicable
- Magic

*There are actually a few things that are production ready right out of the box.

The Good

- Preprocessing
  - segmentation, tokenization, PoS tagging
- Word level processing
  - WordNet, Lemmatization, Stemming, NGram
- Utilities
  - Tree, FreqDist, ConditionalFreqDist
  - Streaming CorpusReader objects
- Classification
  - Maximum Entropy, Naive Bayes, Decision Tree
  - Chunking, Named Entity Recognition
- Parsers Galore!
- Languages Galore!

The Bad

- Syntactic Parsing
  - No included grammar (not a black box)
- Feature/Dependency Parsing
  - No included feature grammar
- The sem package
  - Toy only (lambda-calculus & first order logic)
- Lots of extra stuff
  - papers, chat programs, alignments, etc.

Other Python NLP Libraries

- TextBlob
- SpaCy
- Scikit-Learn
- Pattern
- gensim
- MITIE
- guess_language
- Python wrapper for Stanford CoreNLP
- Python wrapper for Berkeley Parser
- readability-lxml
- BeautifulSoup

NLTK Demo

Working with Text

- Working with Included Corpora
- Segmentation
- Tokenization
- Tagging
- A Parsing Exercise
- Named Entity Recognition

Machine Learning on Text

Machine learning uses instances (examples) of data to fit a parameterized model which is used to make predictions concerning new instances.

In text analysis, what are the instances?

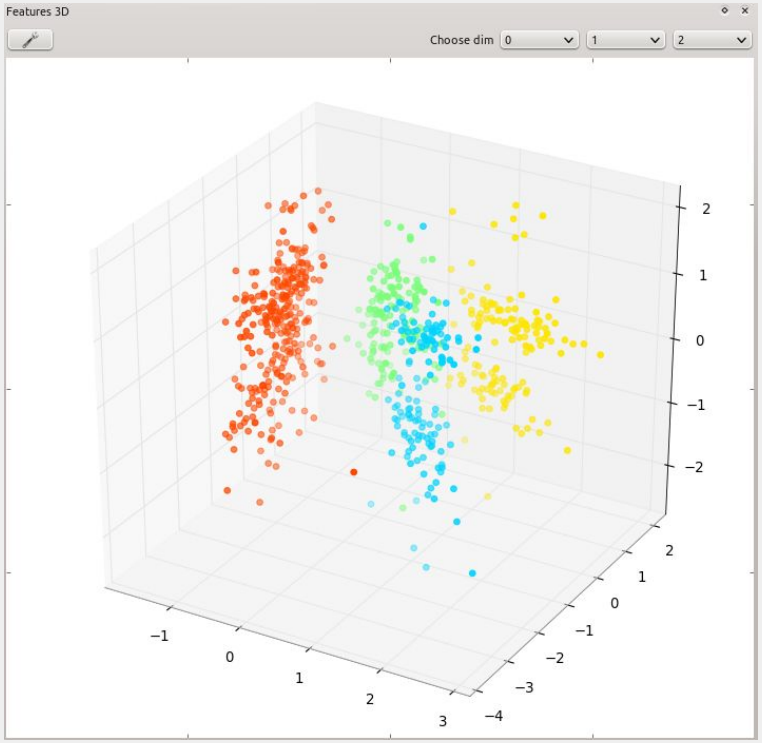
Instances = Documents

(no matter their size)

A corpus is a collection of documents to learn about.

(labeled or unlabeled)

Features describe instances in a way that machines can learn on by putting them into feature space.



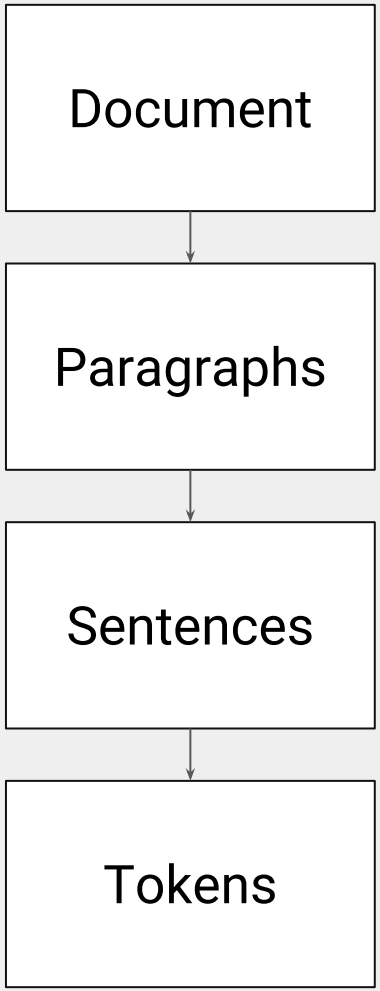
Document Features

Document level features

- Metadata: title, author
- Paragraphs
- Sentence construction

Word level features

- Vocabulary
- Form (capitalization)
- Frequency



Vector Encoding

- Basic representation of documents: a vector whose length is equal to the vocabulary of the entire corpus.
- Word positions in the vector are based on lexicographic order.

The elephant sneezed at the sight of potatoes.

Bats can see via echolocation. See the bat sight sneeze!

Wondering, she opened the door to the studio.

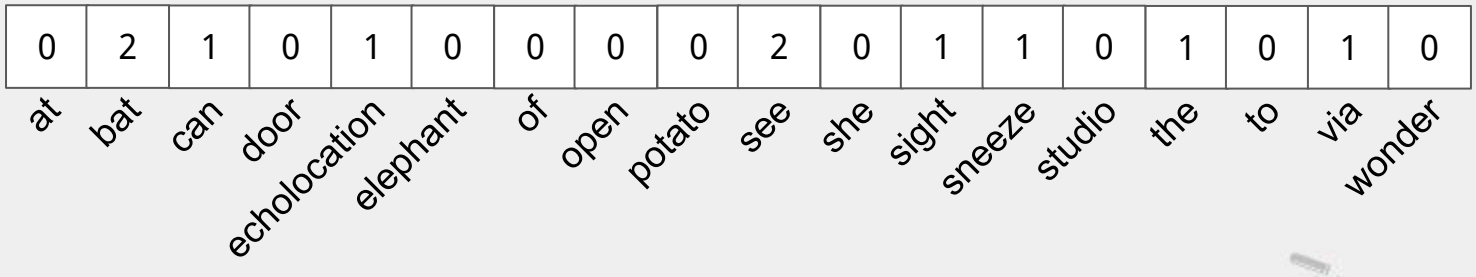
Bag of Words: Token Frequency

- One of the simplest models: compute the frequency of words in the document and use those numbers as the vector encoding.

The elephant sneezed at the sight of potatoes.

Bats can see via echolocation. See the bat sight sneeze!

Wondering, she opened the door to the studio.



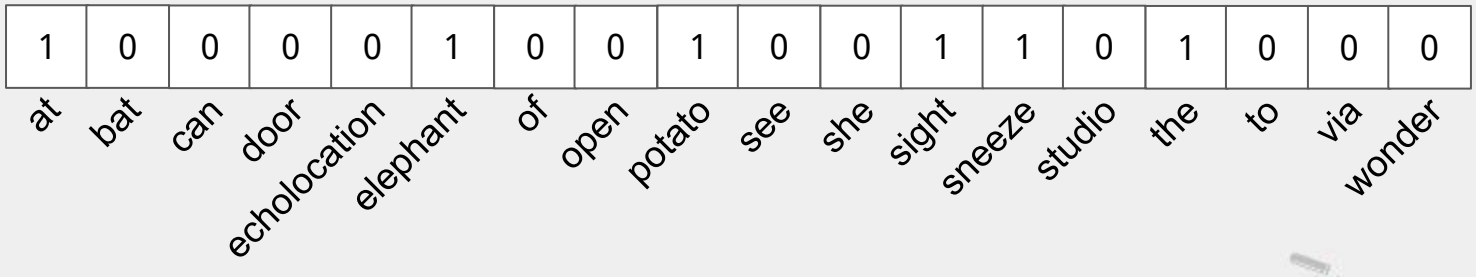
One Hot Encoding

- The feature vector encodes the vocabulary of the document.
- All words are equally distant, so must reduce word forms.
- Usually used for artificial neural network models.

The elephant sneezed at the sight of potatoes.

Bats can see via echolocation. See the bat sight sneeze!

Wondering, she opened the door to the studio.



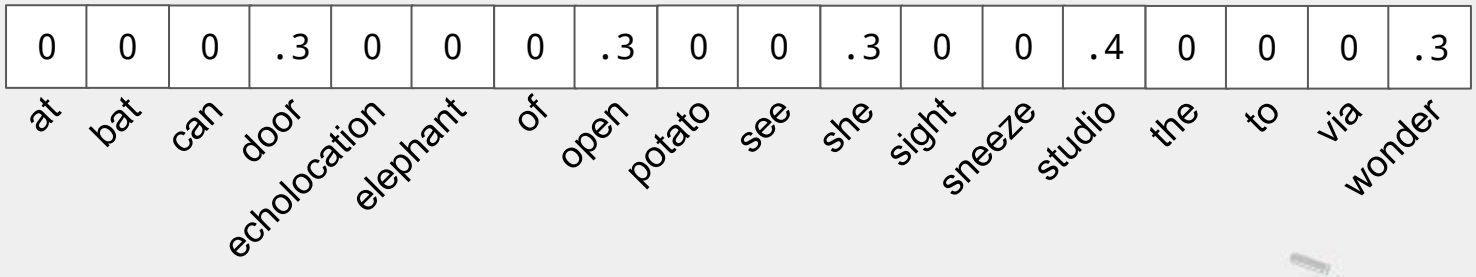
TF-IDF Encoding

- Highlight terms that are very relevant to a document relative to the rest of the corpus by computing the term frequency times the inverse document frequency of the term.

The elephant sneezed at the sight of potatoes.

Bats can see via echolocation. See the bat sight sneeze!

Wondering, she opened the door to the studio.



Pros and Cons of Vector Encoding

Pros

- Machine learning requires a vector anyway.
- Can embed complex representations like TF-IDF into the vector form.
- Drives towards token-concept mapping without rules.

Cons

- The vectors have lots of columns (high dimension)
- Word order, grammar, and other structural features are natively lost.
- Difficult to add knowledge to learning process.

In the end, much of the work for language aware applications comes from domain specific feature analysis; not just simple vectorization.

Classification and Clustering

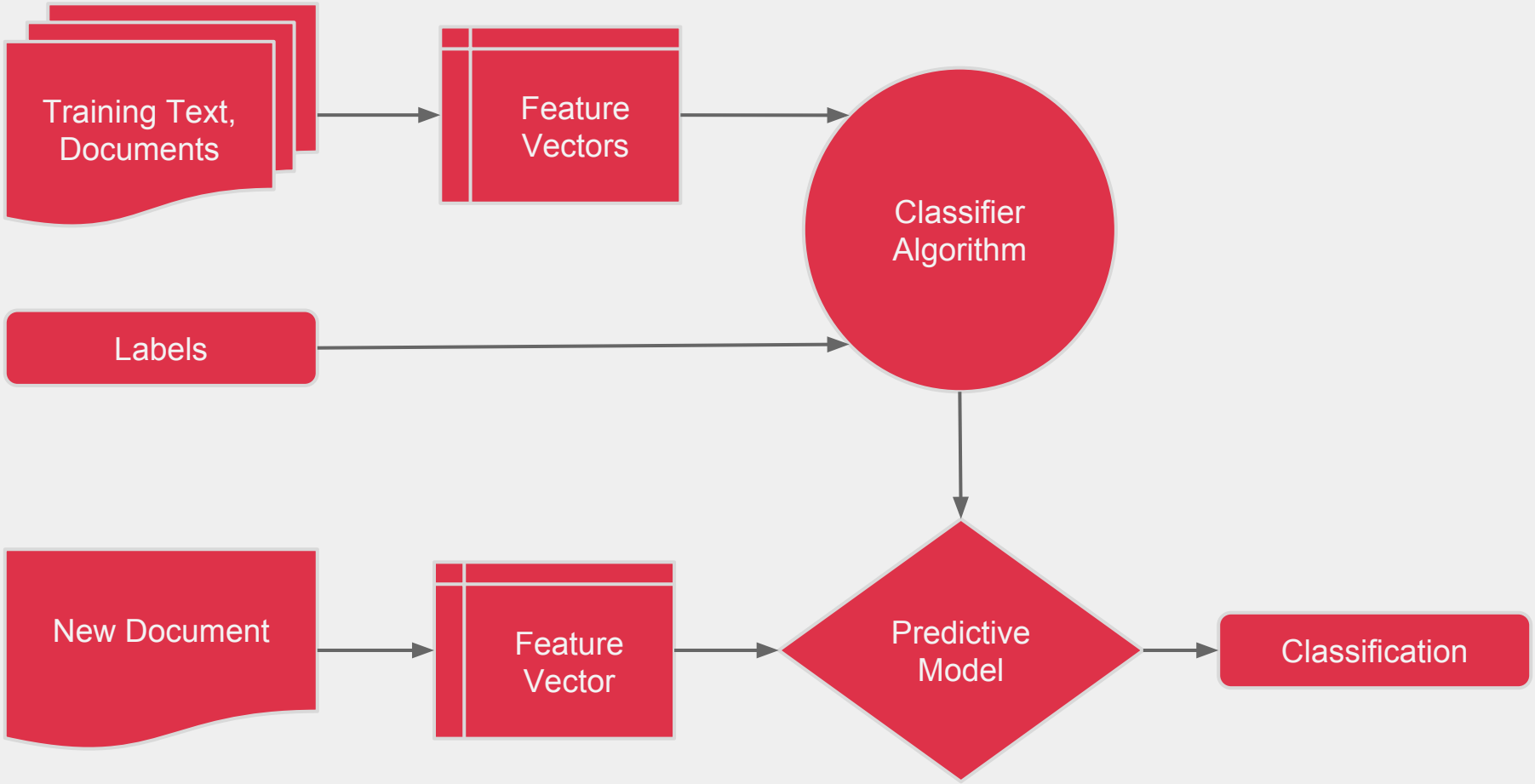
Classification

- Supervised ML
- Requires pre-labeled corpus of documents
- Sentiment Analysis
- Models:
  - Naive Bayes
  - Maximum Entropy

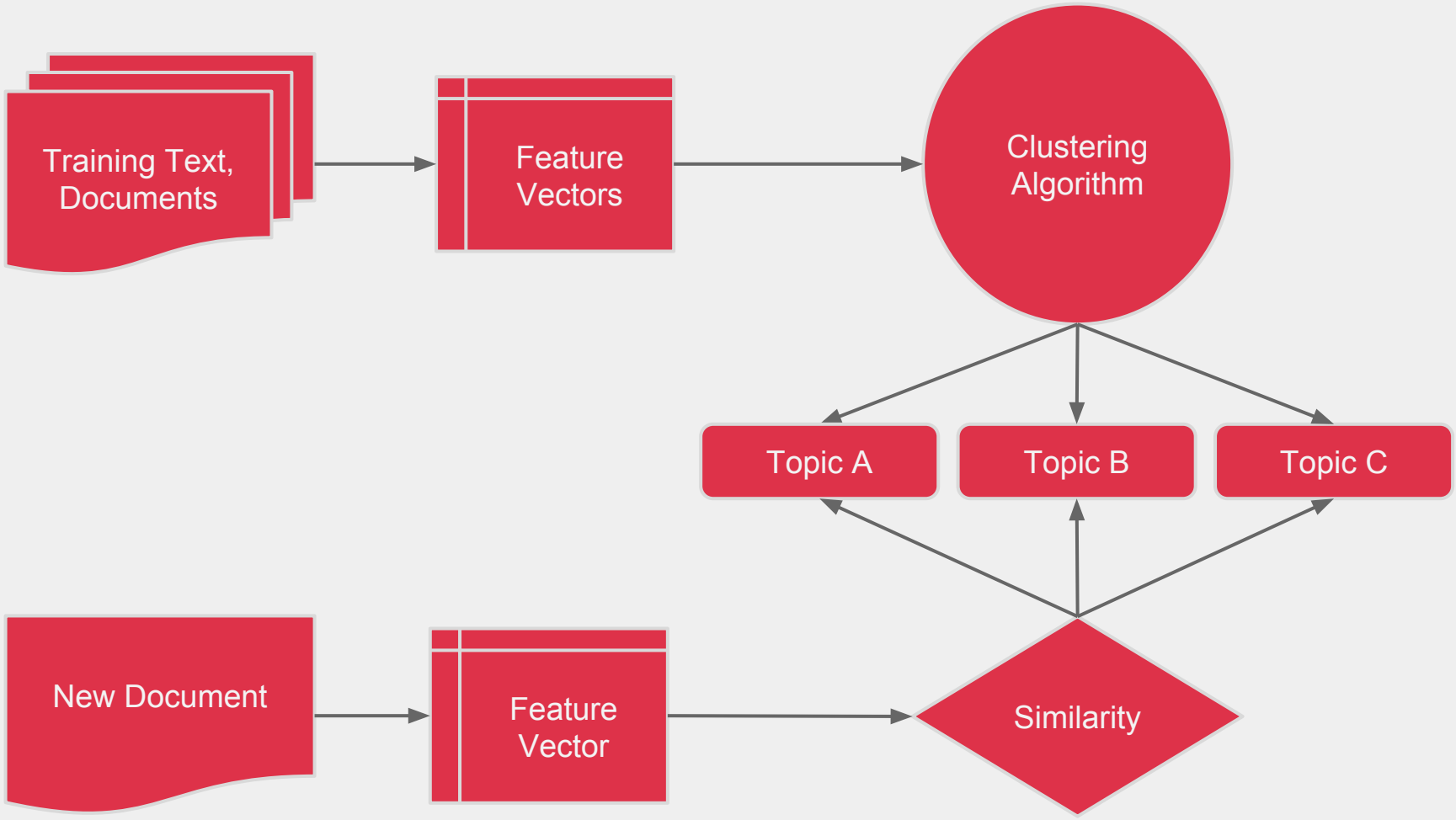
Clustering

- Unsupervised ML
- Groups similar documents together.
- Topic Modeling
- Models:
  - LDA
  - NNMF

Classification Pipeline



Topic Modeling Pipeline



These two fundamental techniques are the basis of most NLP.

It’s all about designing the problem correctly.

Consider an automatic question answering system: how might you architect the application?

Production Grade NLP

gensim

- Access to LDA model
- Good TF-IDF Modeling
- Use word2vec

NLTK
TextBlob

- Text processing
- Lexical resources
- Access to WordNet

- More model families
- Faster implementations
- Pipelines for NLP

Our Task

Build a system that ingests raw language data and transforms it into a suitable representation for creating revolutionary applications.