

Using Text Analysis And Data Modeling To Understand Big Data

Maryam Jahanshahi Ph.D.
Research Scientist

tap Recruit.co

Today:

1. Designing Text Preprocessing Pipelines
2. Transforming Text into Feature Representations
3. Approaches to Classify Documents

Typical Questions when Analyzing Text

Data Resolution:

Are these two pieces of data the same?

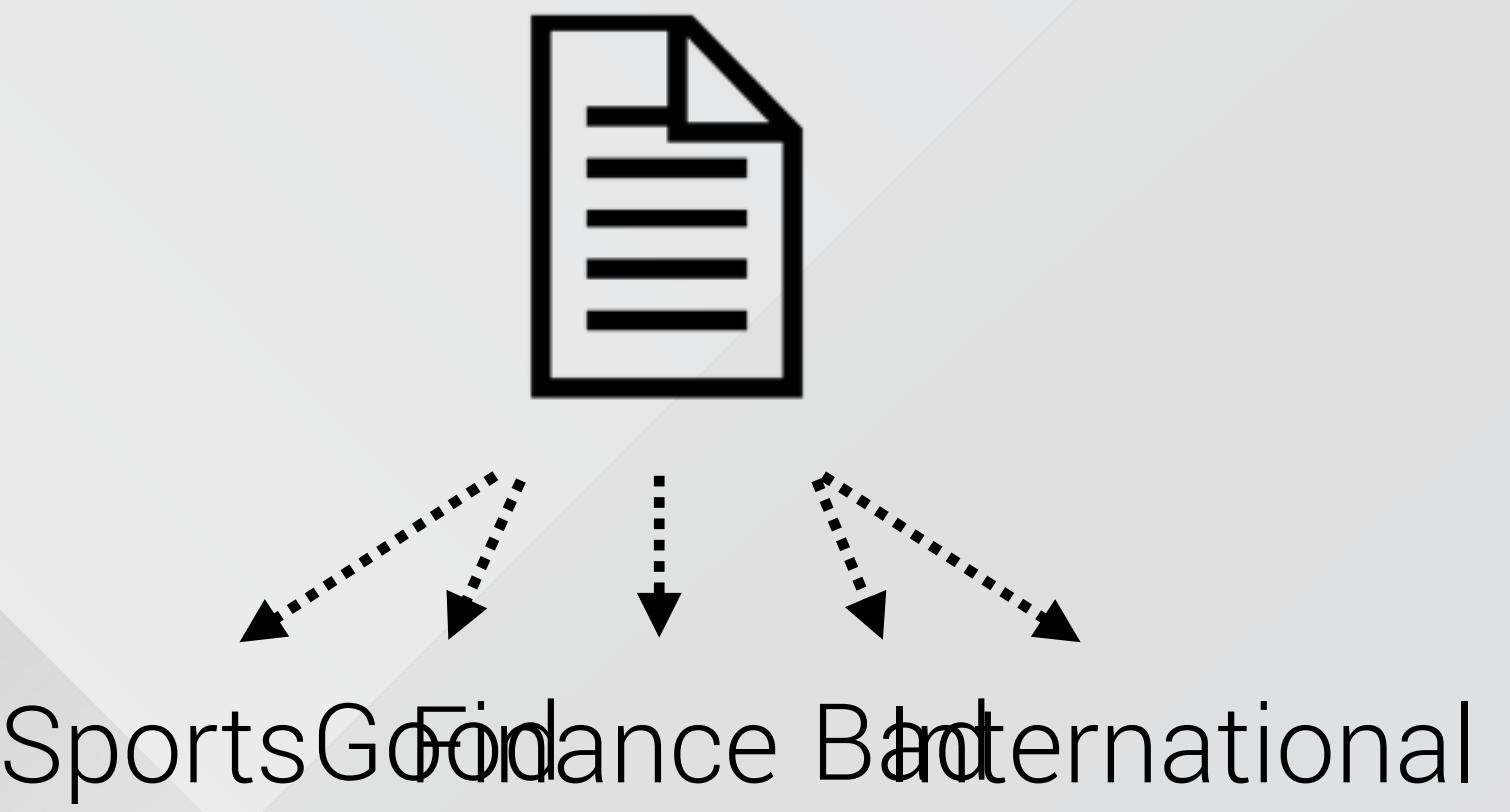
“Software Engineer”
“Software Enginere”

“Software Engineer”
“Sr Software Engineer”

“Software Engineer”
“Computer Programmer”

Document Classification:

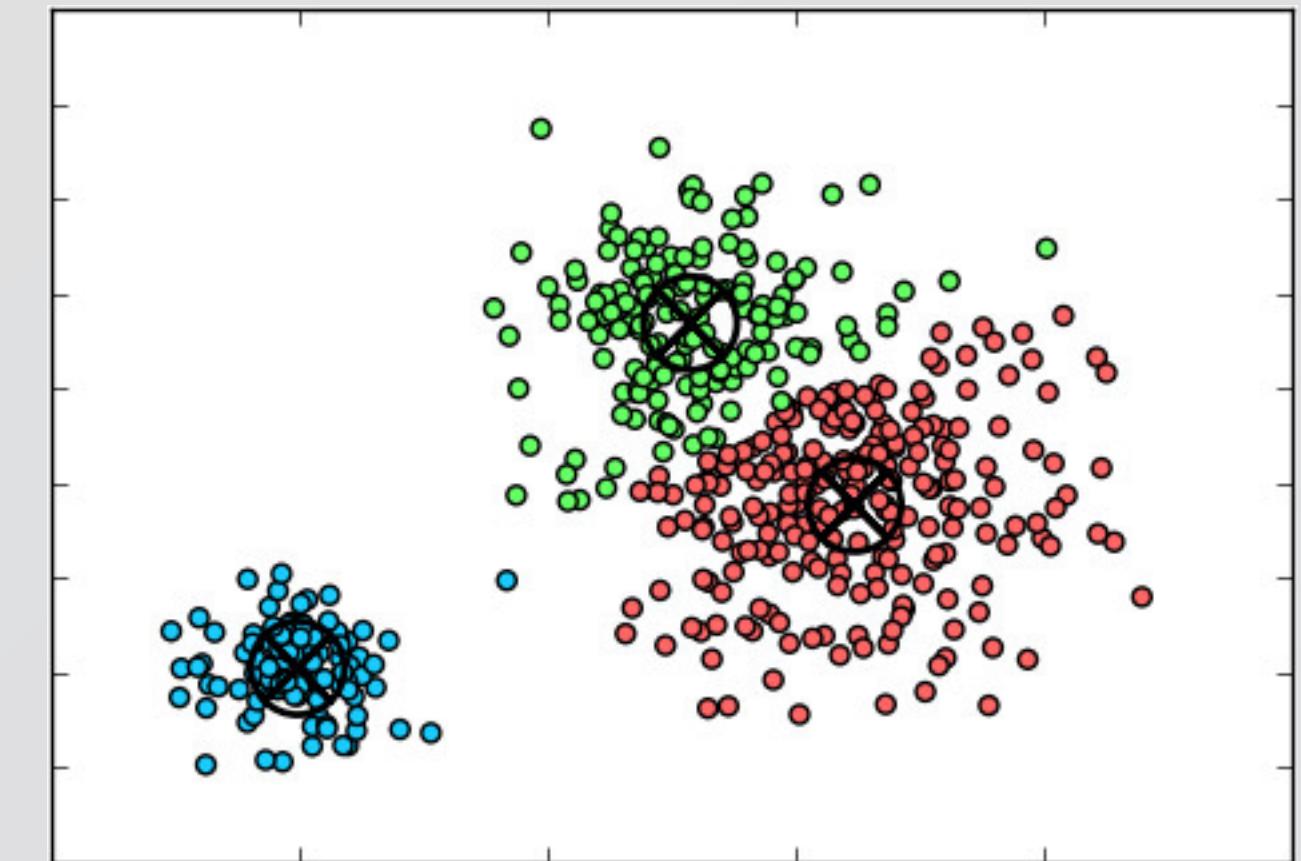
What type of news does this article contain?



Predict Unknown Values

Document Clustering:

What structure does this collection of documents show?



Find Patterns to Describe Data

Approaches to Analyze Text

Text Mining

Goal: Extract Useful Information from Text Data (Overview)

Approaches: Pattern Matching or Matching Structure of Text Data

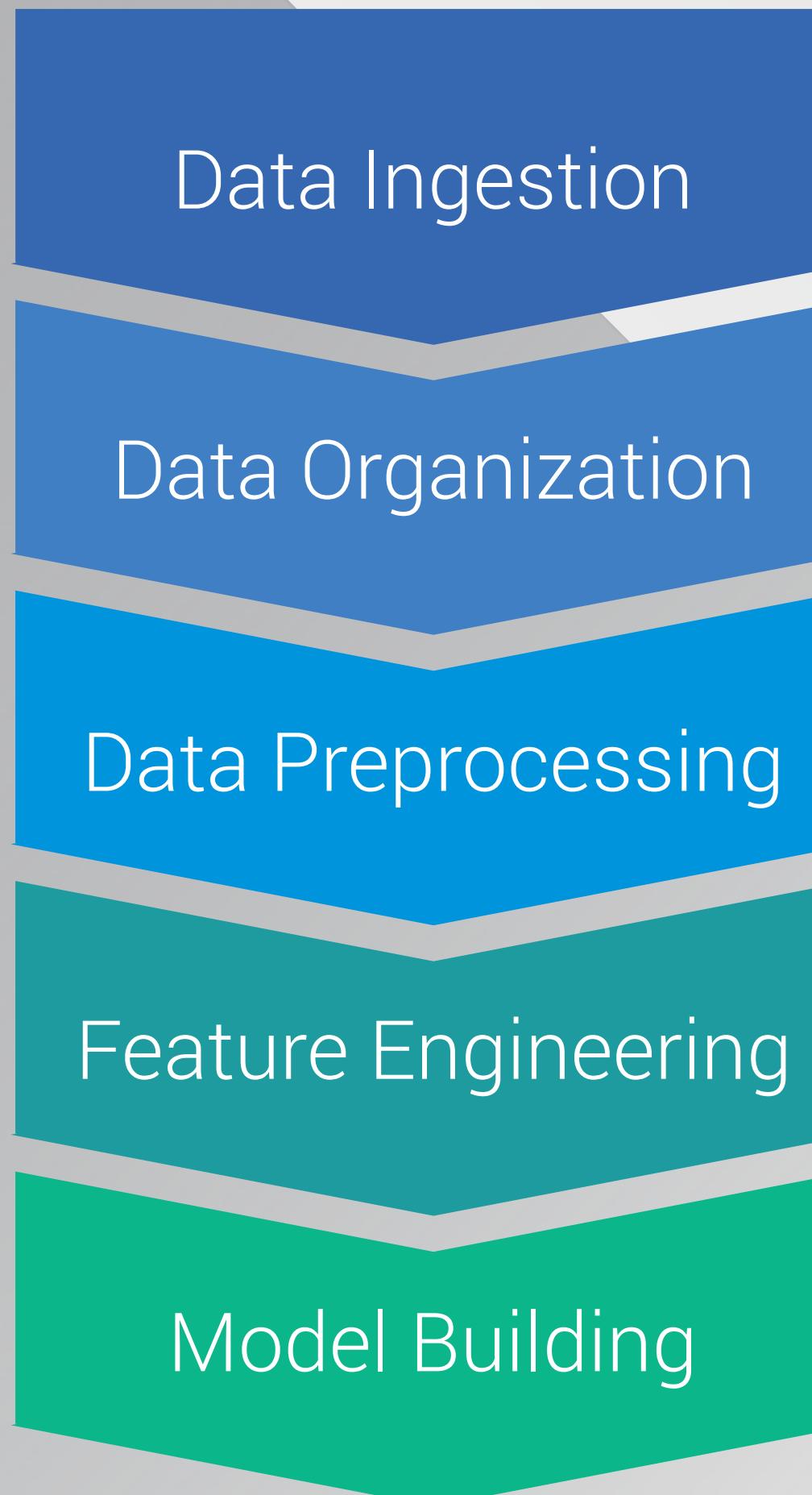
Natural Language Processing

Goal: Understand what the Text is Conveying (Granular)

Approaches: Extracting Semantic Meaning from Text Data

Models used typically include:
Lexical Analysis (Part of Speech tagging)
Named Entity Recognition,
Relationship Extraction and
Semantic Analysis (WordNet, DBpedia)

The Lifecycle of a Text Analysis Project



Resources for Text Analysis Projects

Data Ingestion

Applied Text Analysis with Python

by Benjamin Bengfort, Rebecca Bilbro & Tony Ojeda

Data Organization

Data Preprocessing

Feature Engineering

Model Building

Speech and Language Processing

by Dan Jurafsky & James Martin

Foundations of Statistical NLP

by Chris Manning & Hinrich Schütze

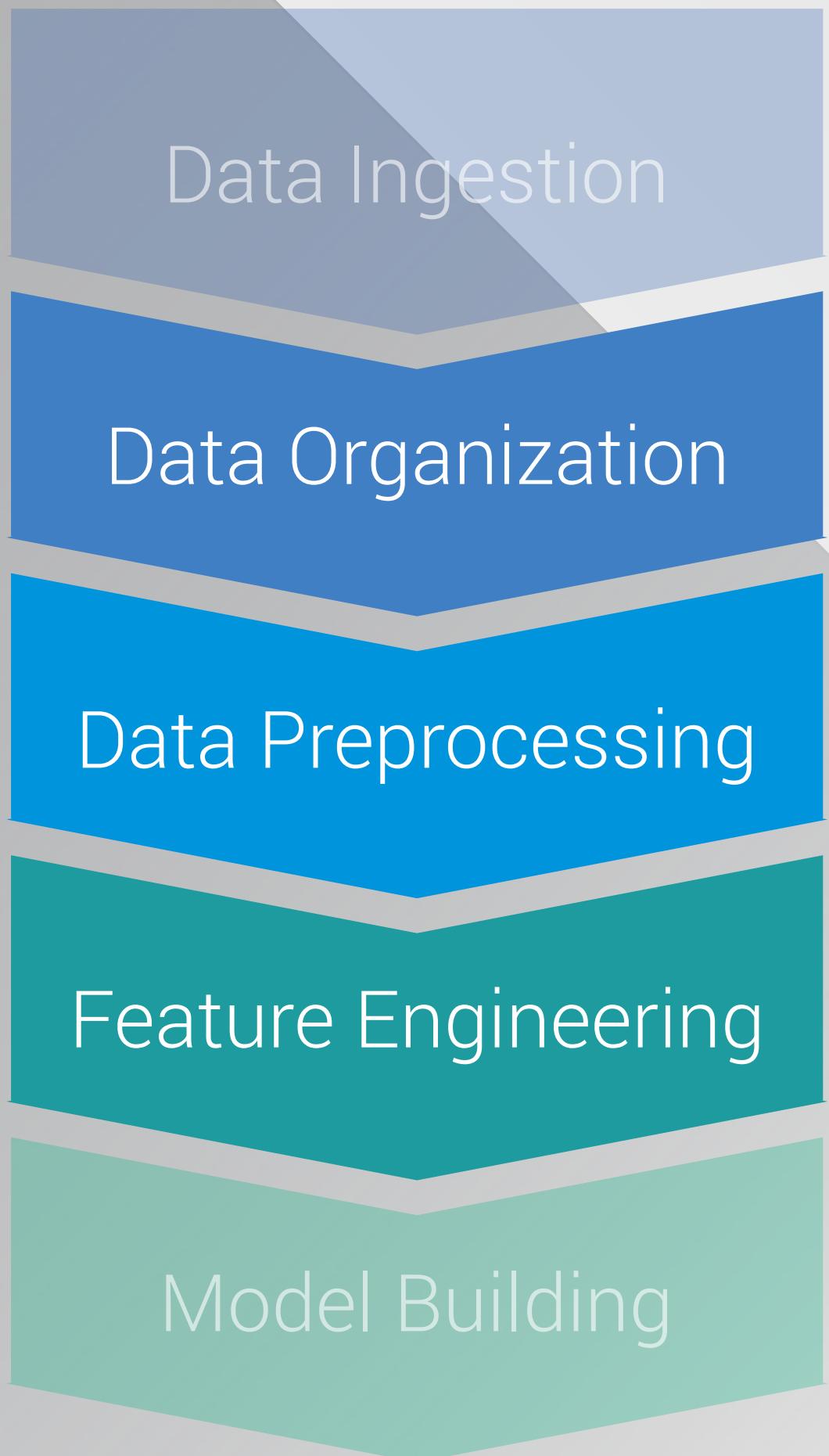
Natural Language Processing with Python

by Steven Bird, Ewan Klein & Edward Loper

Text Mining with R

by Julia Silge & David Robinson

Data Preprocessing is Critical



Alex Gude
@alex_gude

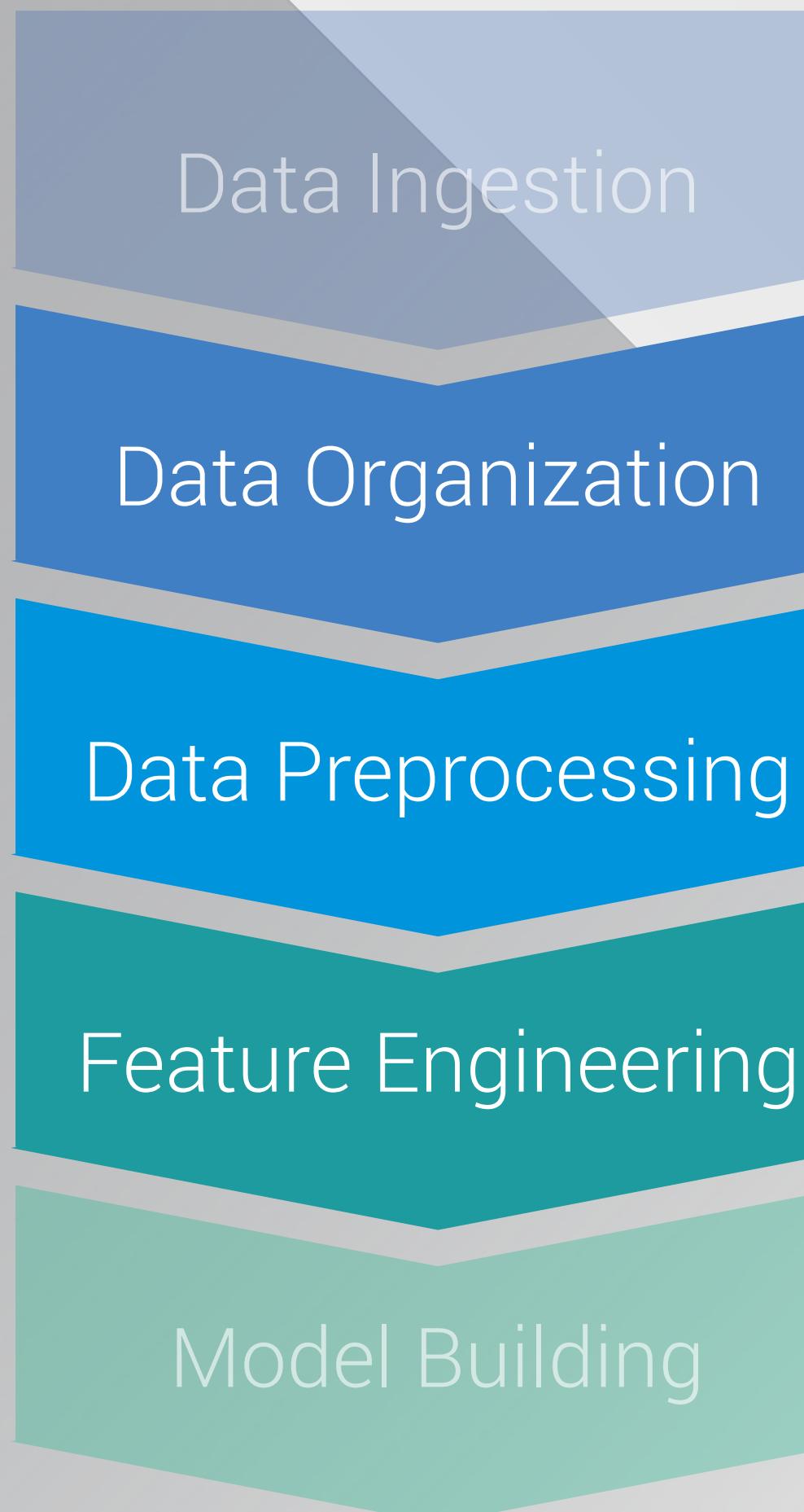
Here is a real use case from work for model improvement and the steps taken to get there:

- Baseline: 53%
- Logistic: 58%
- Deep learning: 61%
- ****Fixing your data: 77%****

Some good ol' fashion "understanding your data" is worth it's weight in hyperparameter tuning!

3:48 PM · Apr 24, 2019 · [Twitter Web Client](#)

The Lifecycle of a Text Analysis Project



Corpus:
A collection
of documents

Document:
Unprocessed
string, typically
associated with
structured data

Segment:
Processed
string
(i.e. sentence,
paragraph etc.)

Token:
Single
processed
data point

Hackathons ('hackathon',
are awesome' NN)

Today:

1. Designing Text Preprocessing Pipelines
2. Transforming Text into Feature Representations
3. Approaches to Classify Documents

Best Practices in Data Organization

Separating Raw Documents from Processed Intermediaries

```
corpus
|_ README.md
|_ raw
|   |_ 01.txt
|   |_ 02.txt
|   |_ 03.txt
|   |_ metadata.json
|_ processed
|   |_ processed.json
|   |_ metadata.json
|_ scripts
```

Processed documents:

Save down processed documents either as JSON objects or in a document database (NoSQL)

Metadata:

Define what has been processed and when in metadata:

- Files
- Words
- Unique Tokens
- Date of latest preprocessing

Advanced Best Practices in Data Organization

Creating a Corpus Reading Module

- Define which files should be loaded and how those files should be loaded.
 - Store these as parameters in README.
 - Regex for file names / formats `\w\.\txt+`
 - Can include a filter list for restricting files

```
import json

def project_reader(self):
    return json.load(self.open("README"))
```

```
corpus
|_ README.md
|_ raw
|   |_ 01.txt
|   |_ 02.txt
|   |_ 03.txt
|   |_ metadata.json
|_ processed
|   |_ processed.json
|   |_ metadata.json
|_ scripts
```

Designing Data Preprocessors

Clean up

Goal: Remove inconsistency between otherwise similar data points

Segmentation

Goal: Split text chunks into data points (i.e. the unit of analysis or evaluation)

Normalization

Goal: Put data points on an equal footing

Designing Data Preprocessors

Clean up

Segmentation

Normalization

General Considerations

- What is the unit or data structure of analysis?
(Tokens vs sentences vs paragraphs vs docs)
- Can the cleanup aid segmentation?

Specific Considerations

- What is the role of punctuation?
- What role do hyphenated words play?
- Will parsing emojis or emoticons be helpful?

Typical Clean Up Pipeline

HTML/XML

< p > We didn't start the fire! </ p >

Python: **Beautiful Soup**
R: **xml2**

Unicode

We didn't start the fire!

Python: **regex**
R: **utf8**

Contractions

We didn't start the fire!

Python: **spaCy**
R: **textclean**

Punctuation

We did not start the fire!

Python: **spaCy**
R: **textclean**

Tokenization

We | did | not | start | the | fire

Python: **spaCy**
R: **tokenizers**

The Order of Operations Matters!

HTML/XML

< p >We didn't start the fire!**< /p >**

Contractions

We didn't start the fire!

Punctuation

We didnu0027t start the fire

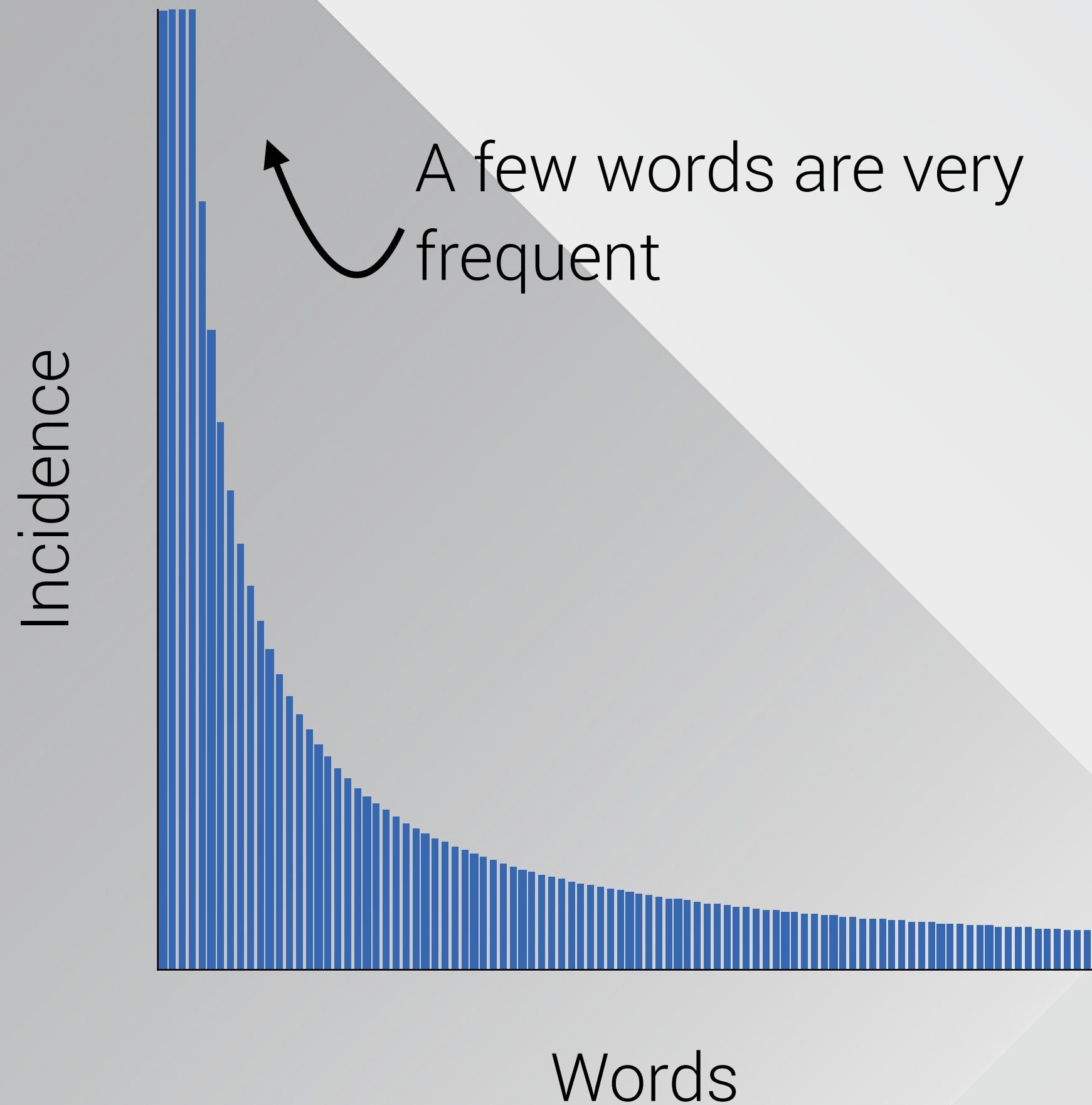
Unicode

We didnu0027t start the fire

Tokenization

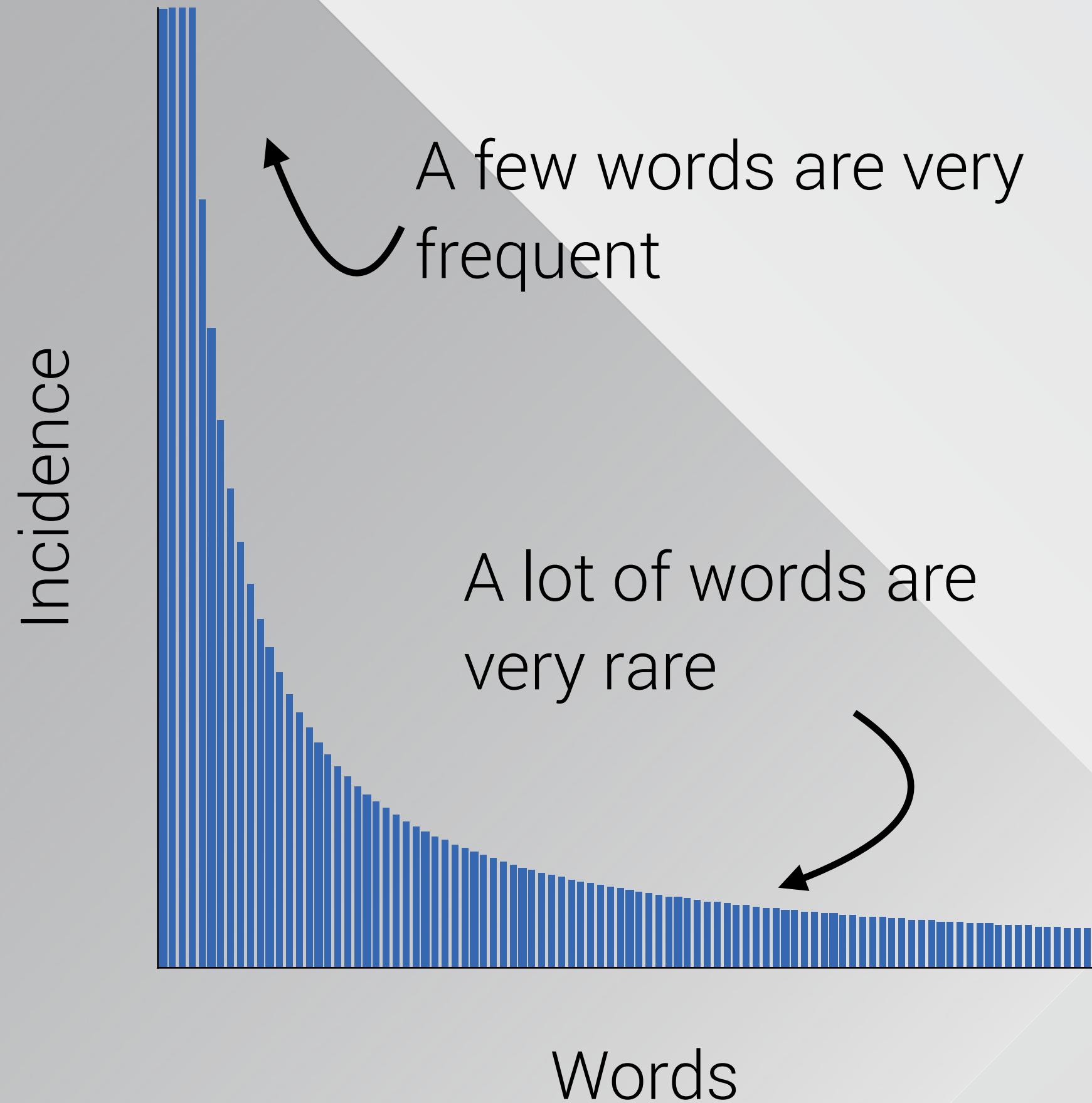
We | didnu0027t | start | the | fire

Word Incidence is Rarely Distributed Normally



- **Stop words:** Removing most frequent words.
 - Standard list with most NLP libraries
 - Make your own artisanal list

Word Incidence is Rarely Distributed Normally



- **Stop words:** Removing most frequent words.
 - Standard list with most NLP libraries
 - Make your own artisanal list
- **Changing cases:**
 - Standard is to convert to lower case
 - Casing may matter for you (e.g. IT vs it)
- **Process numbers:**
 - Standard is to remove all numbers
 - Convert into words via **inflect** library in Python and **textclean** package in R
- **Correct spelling:**
- **Stem or lemmatize words:** Lemmatization is standard

Case Study 1

Add information from a new source to your CRM database, joining on job titles.

The new information comes from a scraped website.

How should we preprocess this data to ensure we correctly resolve duplicate records?

Today:

1. Designing Text Preprocessing Pipelines
2. **Transforming Text into Feature Representations**
3. Approaches to Classify Documents

Feature Representations

Bag of Words representation vectorizes through word counts

“We didn’t start the fire”

baby	0
did	1
doo	0
fire	1
not	1
shark	0
start	1
the	1
we	1

“Baby shark, doo doo doo doo doo”

1	0	6	0	0	1	0	0	0
---	---	---	---	---	---	---	---	---

Feature Representations

Normalized Bag of Words can be instructive

“We didn’t start the fire”

	baby	did	doo	fire	not	shark	start	the	we
Bag of Words	0	1	0	1	1	0	1	1	1
- stop words	0	0	0	1	0	0	1	0	0
+ normalization	0	0	0	0.5	0	0	0.5	0	0

Feature Representation

One Hot Encoding and TFIDF normalize token frequencies

“Baby shark, doo doo doo doo doo”

	baby	did	doo	fire	not	shark	start	the	we
Bag of Words	1	0	6	0	0	1	0	0	0
One Hot Encoding	1	0	1	0	0	1	0	0	0
TFIDF	0.05	0	0.4	0	0	0.2	0	0	0

Overview of Feature Representations Methods

One Hot Encoding	Bag of Words	TF-IDF
Stop word removal?	Stop word removal	No need for stop word removal
None	Document-level normalization	Corpus and document-level normalization

Overview of Feature Representations Methods

Non-distributed			Distributed
One Hot Encoding	Bag of Words	TF-IDF	Embeddings
Stop word removal?	Stop word removal	No need for stop word removal	
None	Document-level normalization	Corpus and document-level normalization	

Word Embeddings capture Semantic Similarities

Statistical modeling through software (e.g. SPSS) or programming language (e.g. **Python**)

Context

Word

Experience in **Python**, Java or other object-oriented programming languages

Context

Word

Context

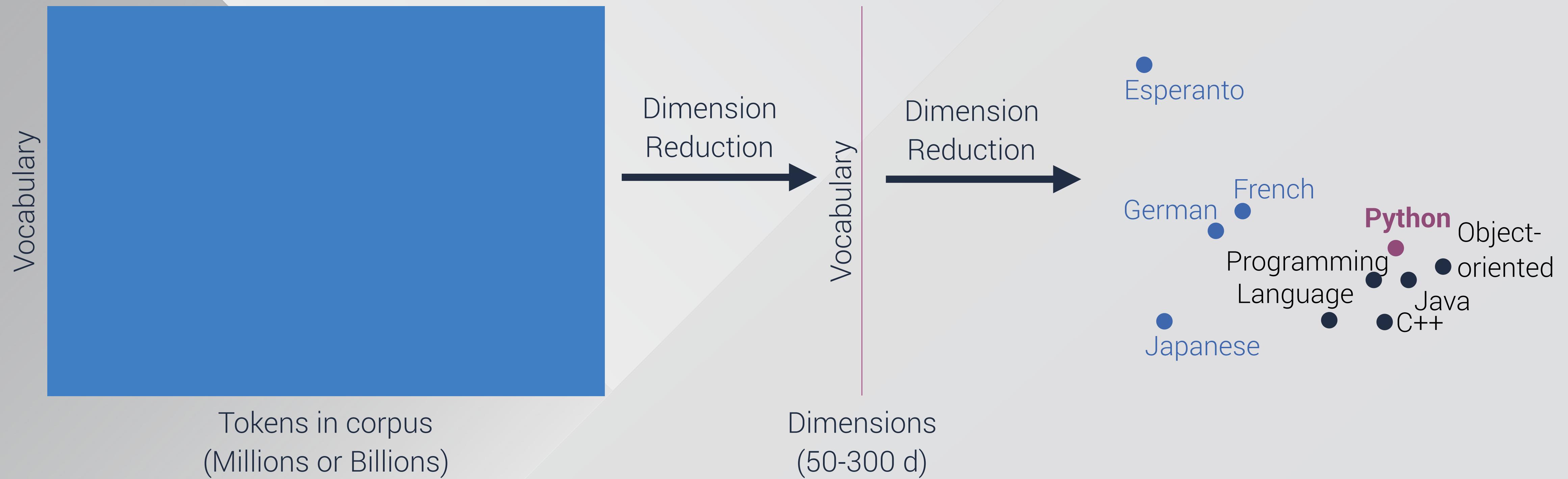
Proficiency programming in **Python**, Java or C++.

Context

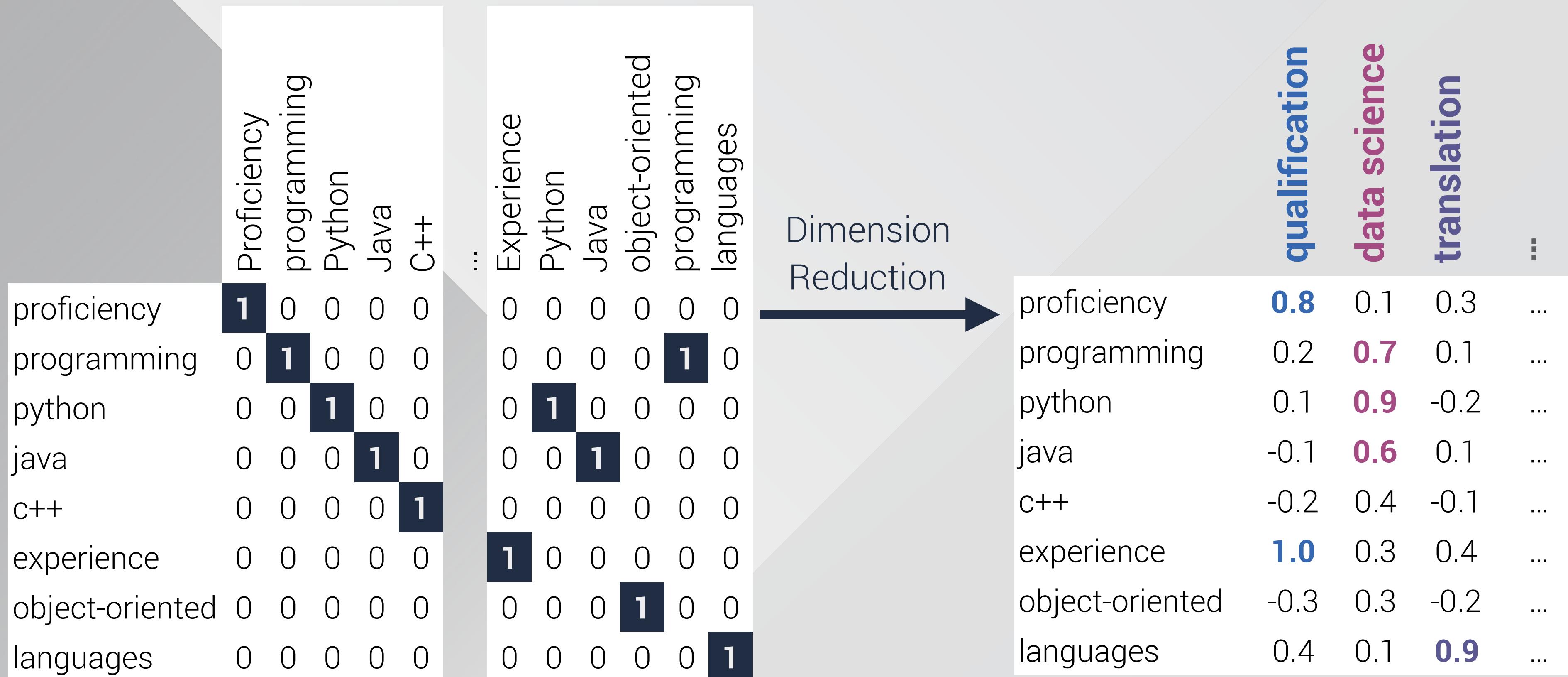
Word

Context

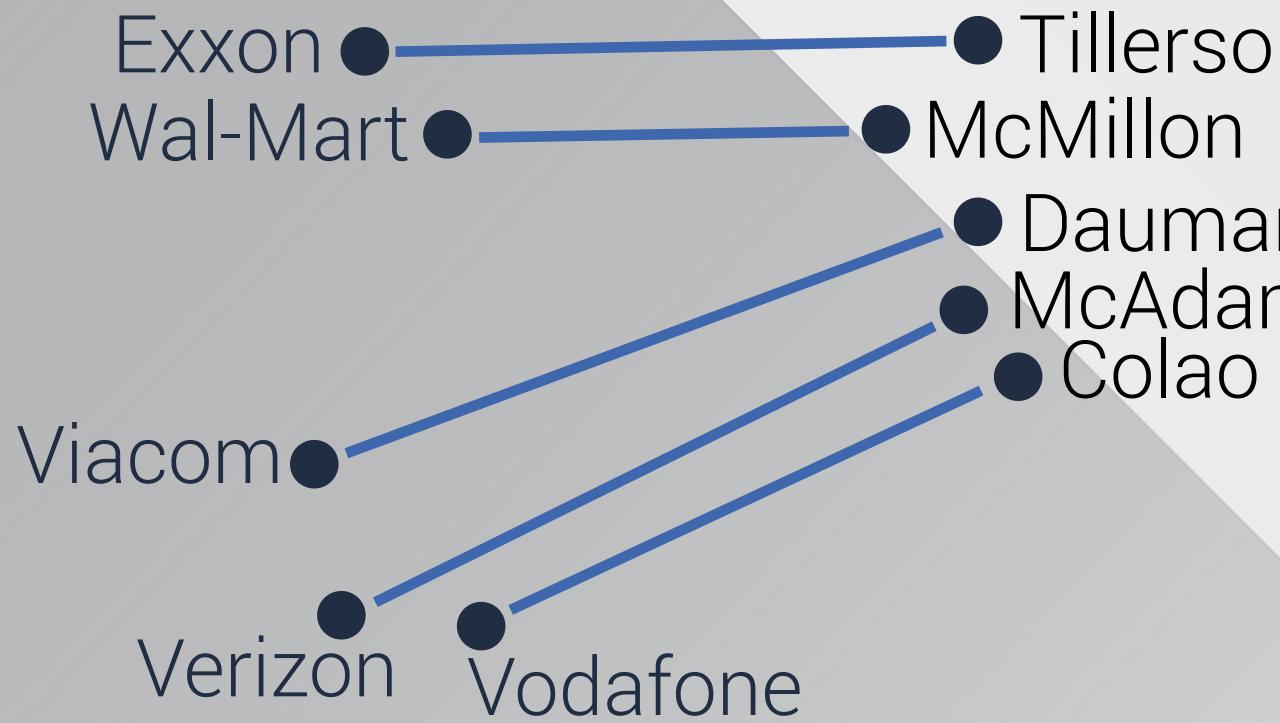
A Simplified Representation of Word Vectors



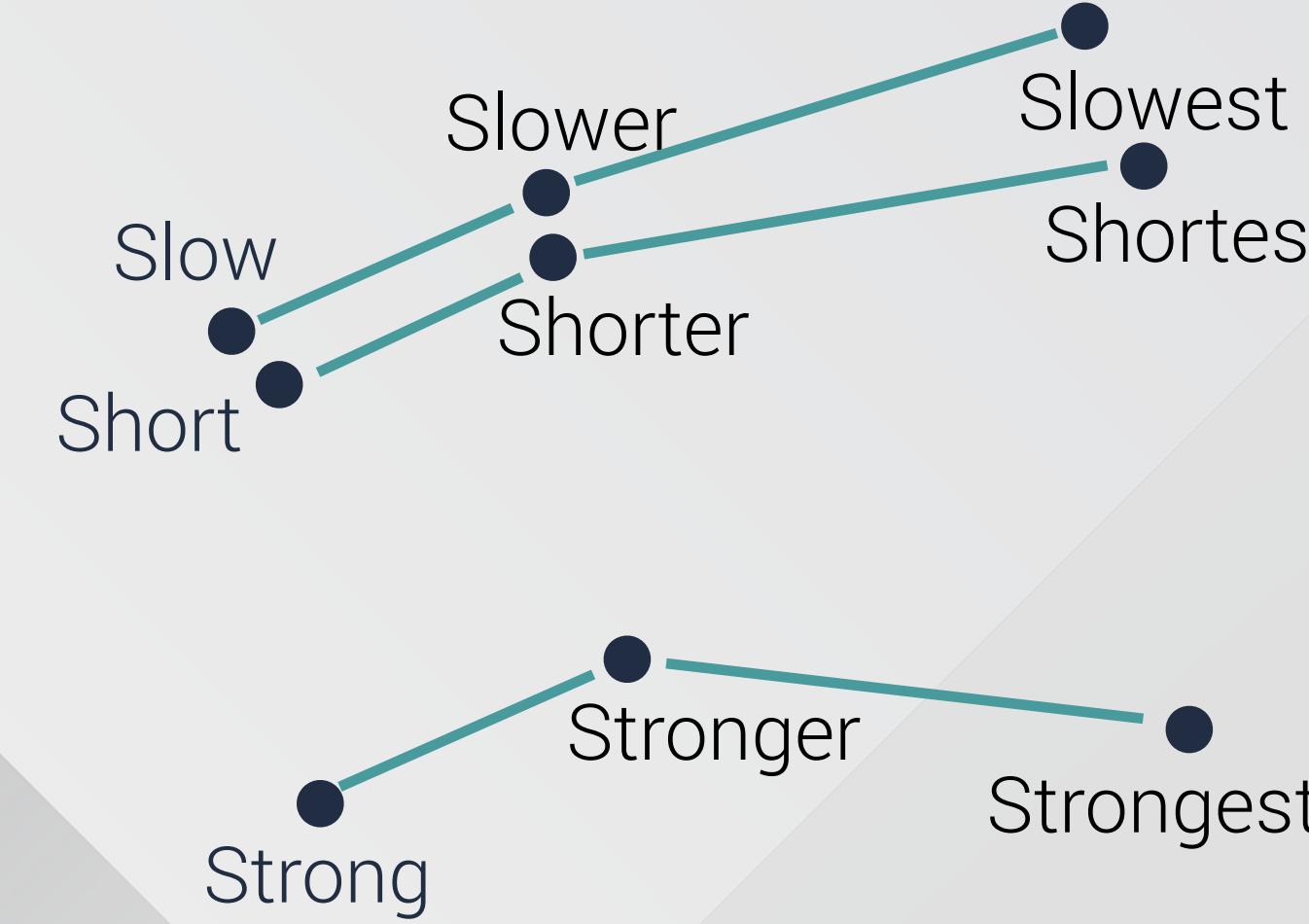
A Simplified Representation of Word Vectors



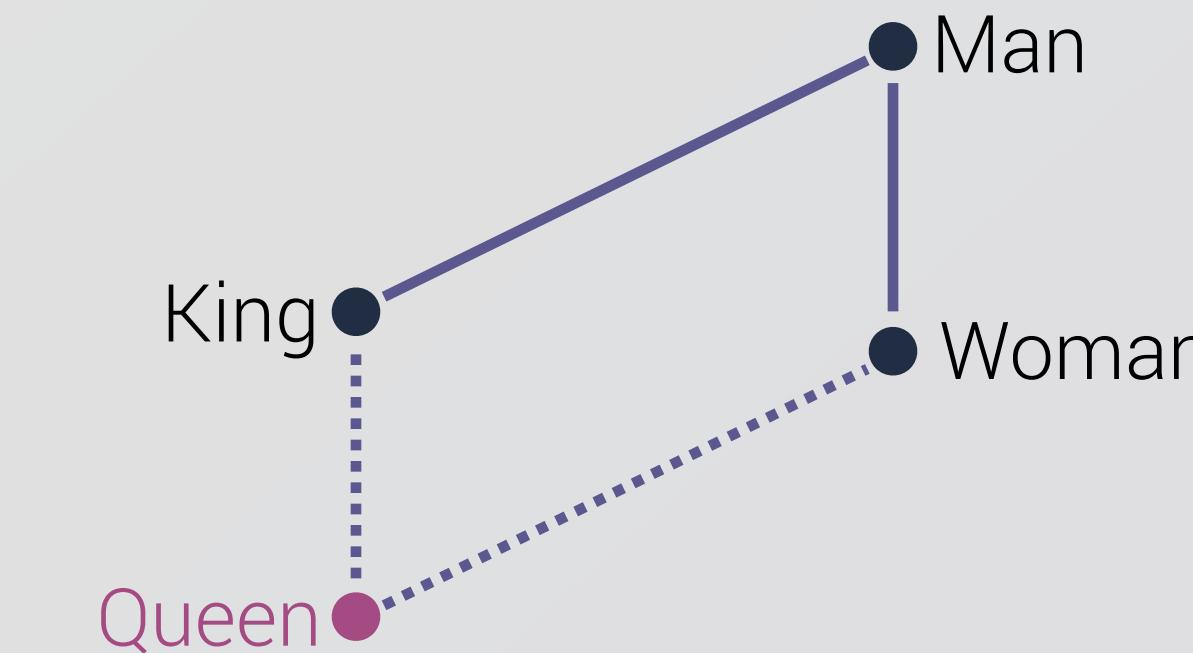
Word Embeddings capture Entity Relationships



Hierarchies

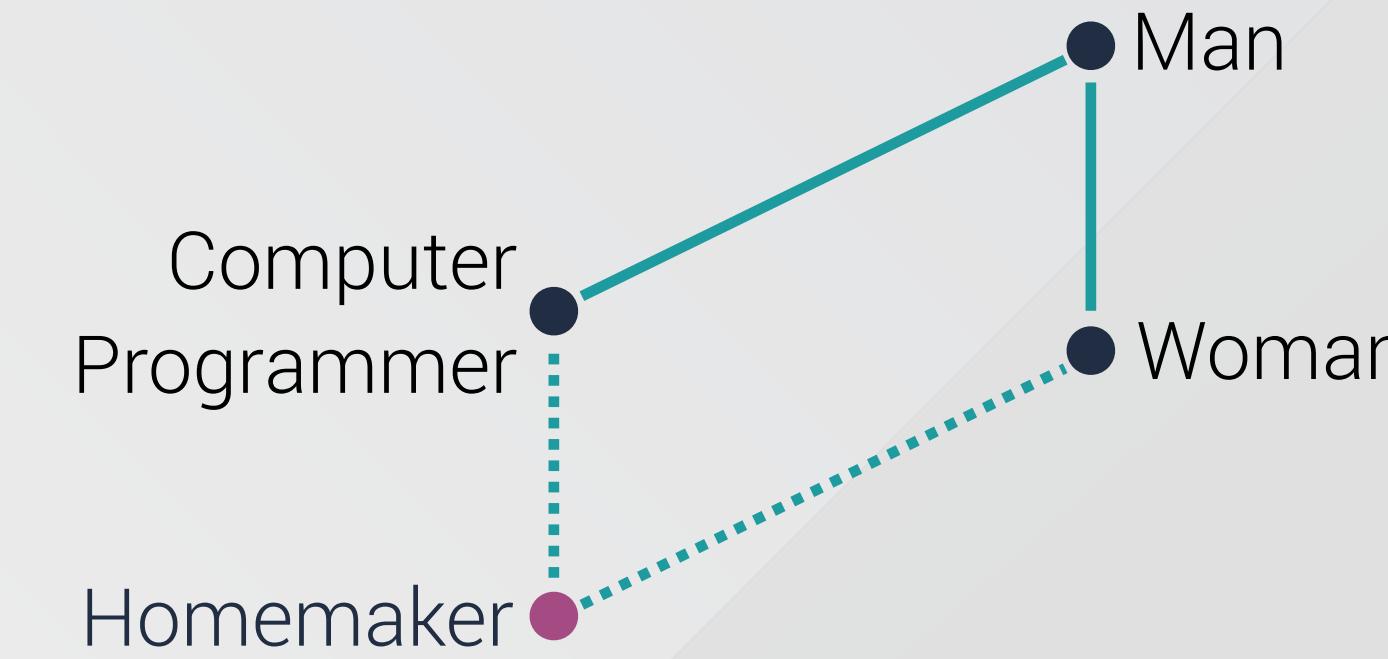
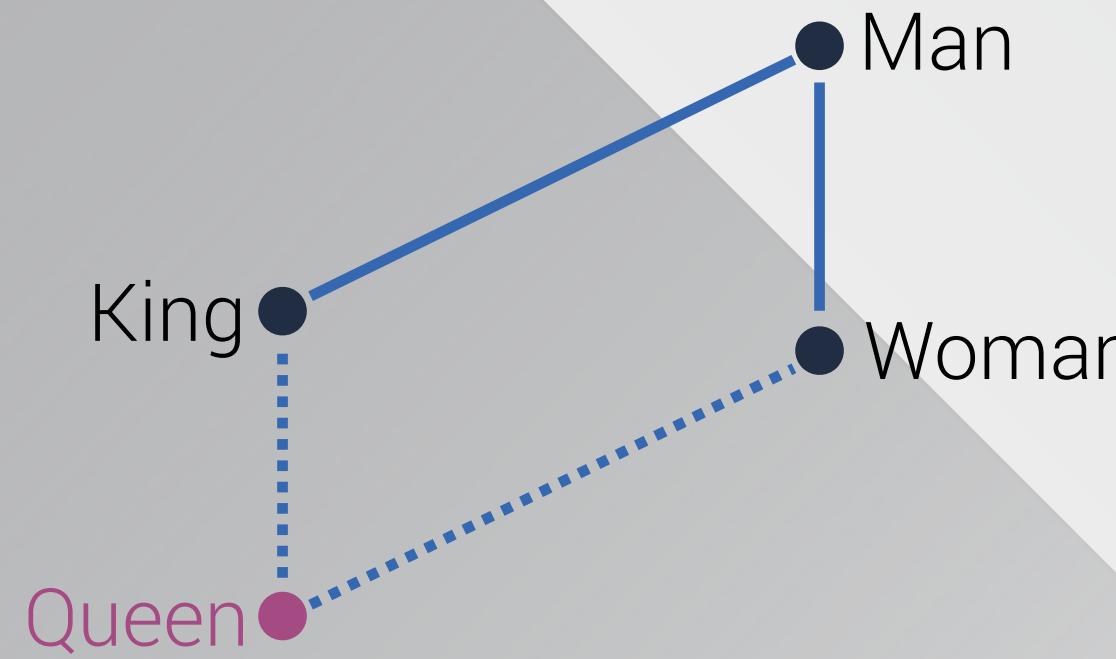


Comparatives and Superlatives



Man :: King as Woman :: ?

Embeddings reflect Cultural Bias in Corpora



Adapted from Bolukbasi et al., [arXiv: 1607:06520](#).

Overview of Feature Representations Methods

Non-distributed		Distributed	
One Hot Encoding	Bag of Words	TF-IDF	Embeddings
Stop word removal?	Stop word removal	No need for stop word removal	Stop word removal?
None	Document-level normalization	Corpus and document-level normalization	Context and corpus-level normalization

Overview of Feature Representations Methods

Non-distributed		Distributed	
One Hot Encoding	Bag of Words	TF-IDF	Embeddings
Stop word removal?	Stop word removal	No need for stop word removal	Stop word removal?
None	Document-level normalization	Corpus and document-level normalization	Context and corpus-level normalization
All tokens are equidistant		Distance \propto token similarity	
High dimensionality, extremely sparse		Lower dimensionality	

Case Study 2

Our marketing department has been running ad campaigns on different social media platforms.

They want to understand which characteristics are shared by the ads that are high performers.

What would we advise them?

Case Study 2

High Performing Ads

How different are the
messages?

Substantial

Subtle

Today:

1. Designing Text Preprocessing Pipelines
2. Transforming Text into Feature Representations
- 3. Approaches to Classify Documents**

Tasks in Document Analysis

Descriptive Approaches

Find patterns to describe data

Approach: Unsupervised Learning
Class labels of data is unknown. Given a set of measurements, goal is to identify some clusters in the data.

Examples: Clustering, Summarization

Predictive Approaches

Predict unknown values from a model using known baselines

Approach: Supervised Learning
Generate a model using training data which has a set of labels which indicate class of observations. Model classifies new data.

Examples: Classification, Ranking, Regression etc.

Overview of Document Categorization

Extracting semantic structure from numeric features

Topic Modeling

What are the topics that occur in a collection of documents?

Every document is a mixture of topics

Every topic is a mixture of words

Unsupervised Dimension Reduction
(LDA / LSA)

Document Classification

Which class does document X belong to?

Every document belongs to a single class

The presence or absence of a subset of words impacts the classification

Supervised ML Algorithms
Regex (Standard or Artisanal)

Case Study 3

Our colleagues have conducted a survey, including a lot of questions that had free-form answers. They need help extracting some insights from these answers.

How would you advise them to analyze the data?

Case Study 3

Extracting Structure from Text Answers

What is the scope of
the answers?



Limited

“What is your
job title?”

Broad

“Any feedback
for us?”



What are you
trying to analyze?



Sentiment

Topics

Typical Questions when Analyzing Text

Data Resolution:

Are these two pieces of data the same?

“Software Engineer”
“Software Enginere”

“Software Engineer”
“Sr Software Engineer”

“Software Engineer”
“Computer Programmer”

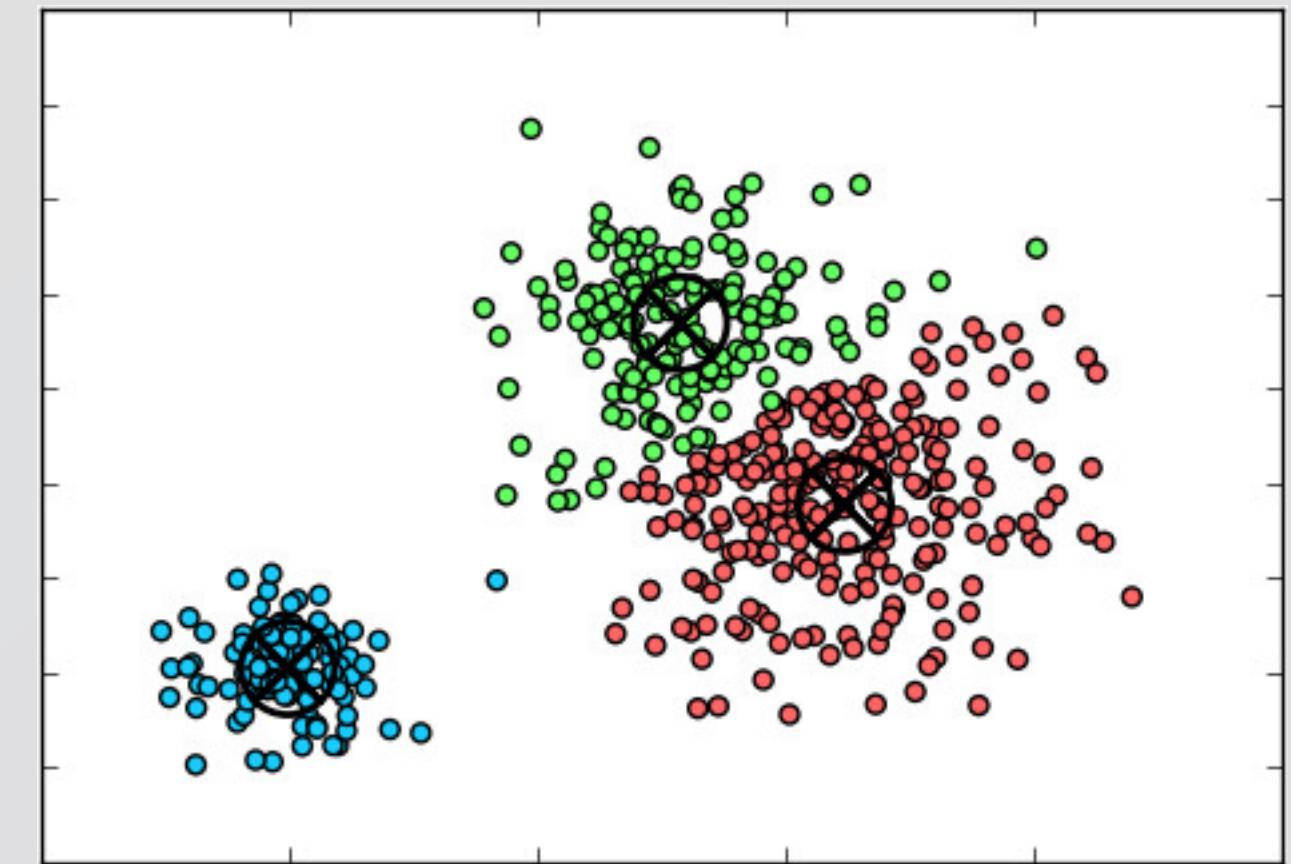
Document Classification:

What type of news does this article contain?



Document Clustering:

What structure does this collection of documents show?



Takeaways:

1. Text Preprocessing

Goal: To put data points on an equal(ish) footing
Cleanup / Segmentation / Normalization
Stopwords / Cases / Spelling / Lemmatization

2. Feature Representation of Text

Goal: To transform text into vectorized representations for downstream
Bag of Words / TFIDF / One-Hot Encoding / Embeddings

3. Document Classification

Goal: To determine whether a document is similar or different to others
Topic Modeling / Document Classification

Thank you ERP Hackathon!

Maryam Jahanshahi Ph.D.
Research Scientist

 @mjahanshahi
 maryam-j

tap Recruit.co