



—  
**BC COMS 2710:**  
**Computational Text Analysis**

**Lecture 6 – Bag of Words**



- Homework 01
  - Due tonight
  
- Readings:
  - Reading 02 – link course site, due Sunday
  
- Week 2 Tutorials:
  - 2.1 – Tokenization, lemmatization, stopwords, etc
    - Based on yesterday's lecture
  - 2.2 – Exploring dictionary-based methods
    - Based on Wednesday's and Thursday's lecture



- Tokenization
- Lemmatization
- Stemming
- Stopwords
- Part of Speech
- Dependency Parsing
- Named Entities



—  
**Zipf's law**  
—





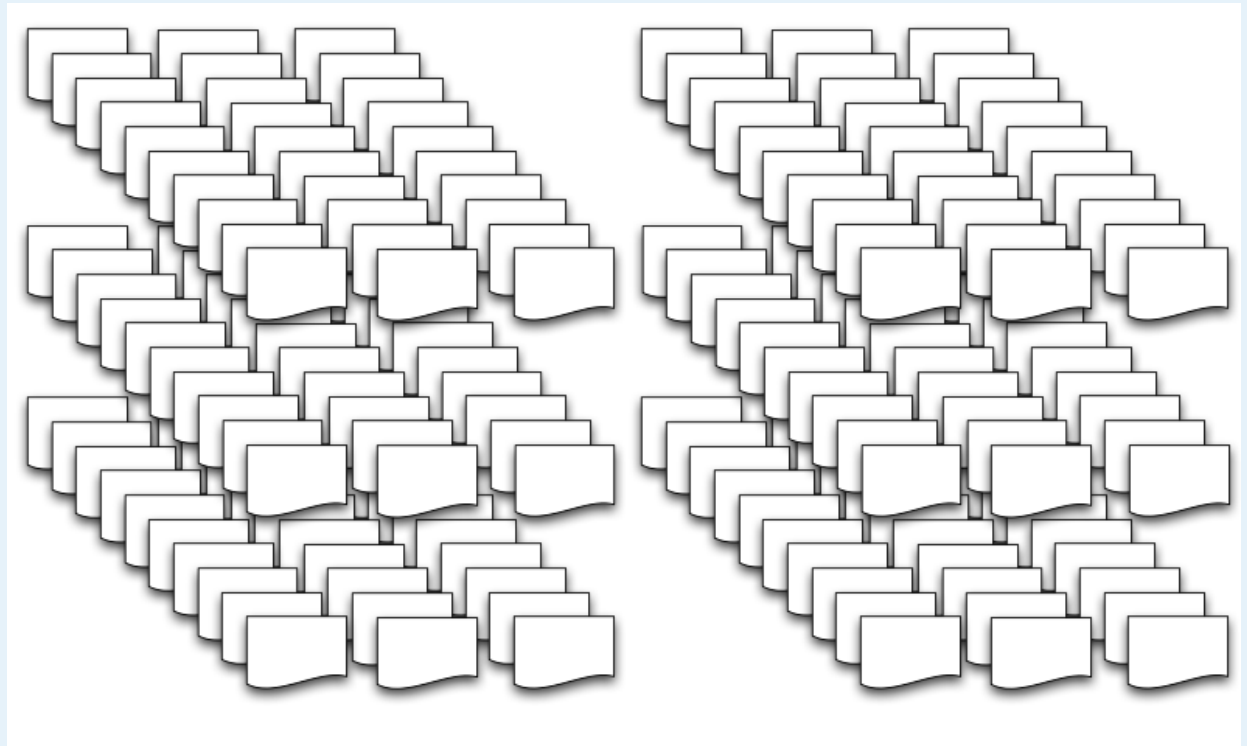
—

# Documents & Corpora

—

## ■ Corpus:

- A collection of documents
- *Corpora* – plural of corpus





- **Document:**
  - Unit of text of interest
  - Often represents one data point
  
- **Examples:**
  - Book
  - Chapter
  - News article
  - Tweet
  - Product Review
  - ....

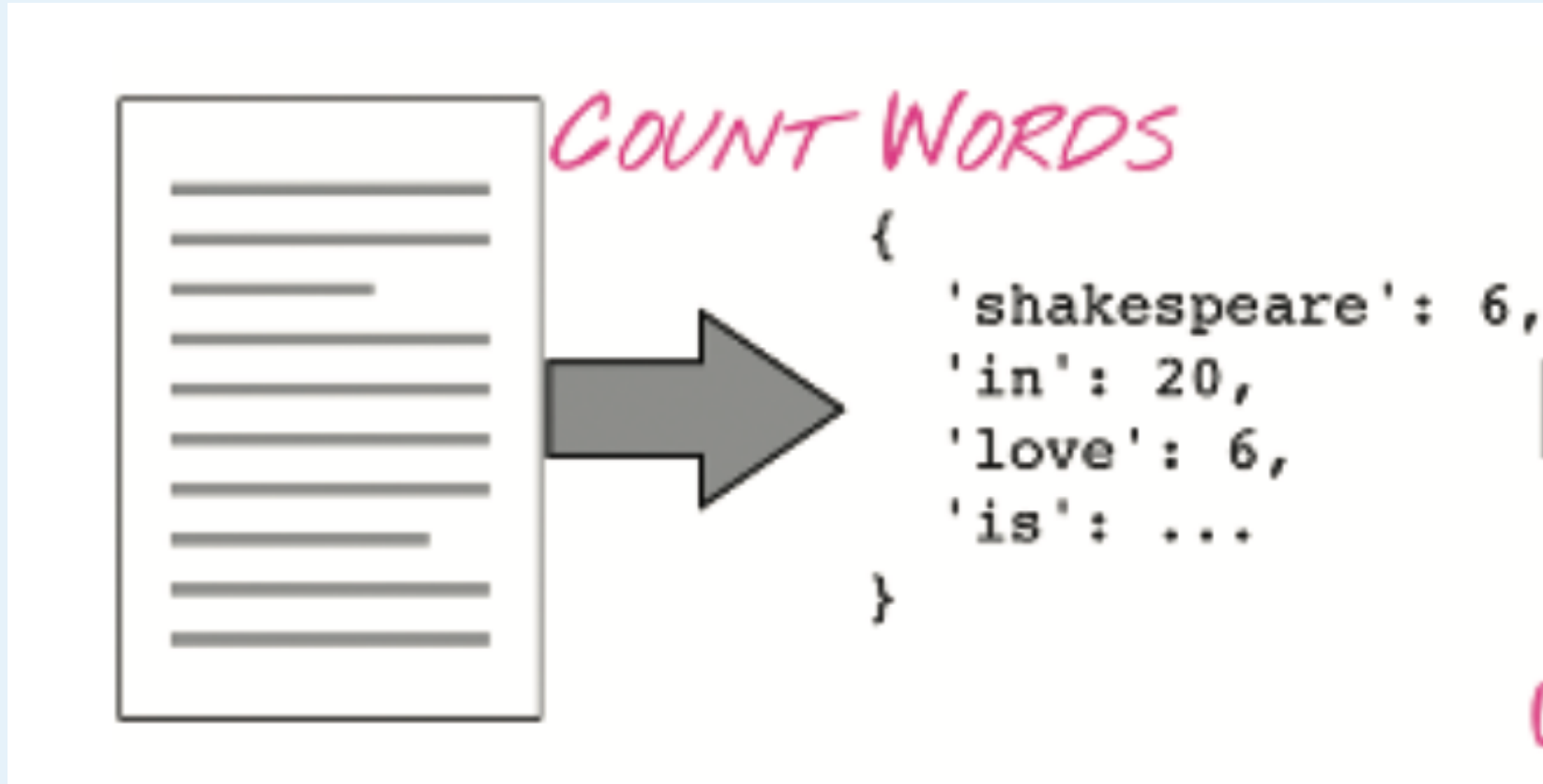


—

# How do we represent documents?

—





Often called *Bag of Words*

# Bag of Words – Start with document

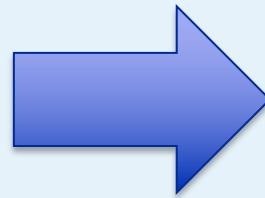


Very good drama although it appeared to have a few blank areas leaving the viewers to fill in the action for themselves. I can imagine life being this way for someone who can neither read nor write. This film simply smacked of the real world: the wife who is suddenly the sole supporter, the live-in relatives and their quarrels, the troubled child who gets knocked up and then, typically, drops out of school, a jackass husband who takes the nest egg and buys beer with it. 2 thumbs up... very very very good movie.

# Bag of Words – Break document into words



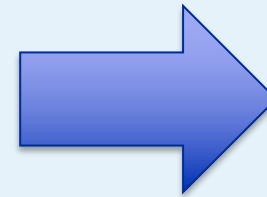
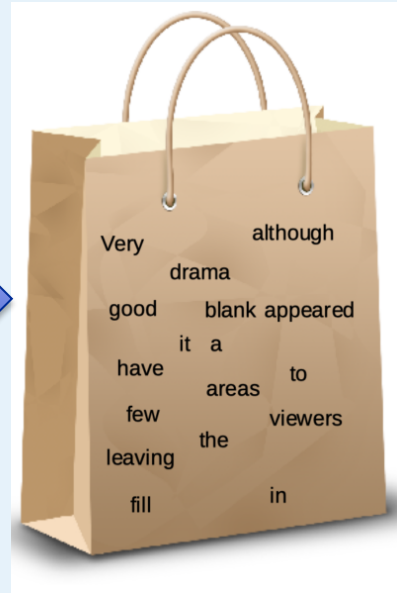
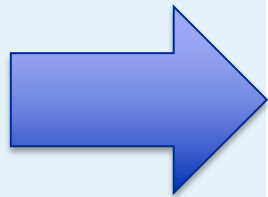
Very good drama although it appeared to have a few blank areas leaving the viewers to fill in the action for themselves. I can imagine life being this way for someone who can neither read nor write. This film simply smacked of the real world: the wife who is suddenly the sole supporter, the live-in relatives and their quarrels, the troubled child who gets knocked up and then, typically, drops out of school, a jackass husband who takes the nest egg and buys beer with it. 2 thumbs up... very very very good movie.



# Bag of Words – compute word counts



Very good drama although it appeared to have a few blank areas leaving the viewers to fill in the action for themselves. I can imagine life being this way for someone who can neither read nor write. This film simply smacked of the real world: the wife who is suddenly the sole supporter, the live-in relatives and their quarrels, the troubled child who gets knocked up and then, typically, drops out of school, a jackass husband who takes the nest egg and buys beer with it. 2 thumbs up... very very very good movie.



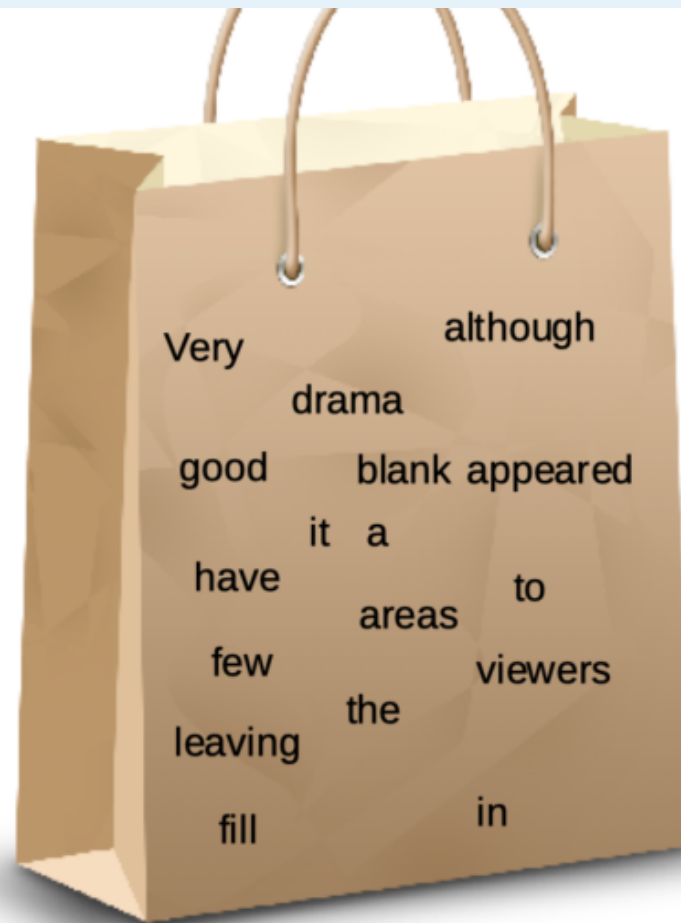
('the', 8),  
(',', 5),  
('very', 4),  
('.', 4),  
('who', 4),  
('and', 3),  
('good', 2),  
('it', 2),  
('to', 2),  
('a', 2),  
('for', 2),  
('can', 2),  
('this', 2),  
('of', 2),  
('drama', 1),  
('although', 1),  
('appeared', 1),  
('have', 1),  
('few', 1),  
('blank', 1)  
.....



# Bag of Words



Very good drama although it appeared to have a few blank areas leaving the viewers to fill in the action for themselves. I can imagine life being this way for someone who can neither read nor write. This film simply smacked of the real world: the wife who is suddenly the sole supporter, the live-in relatives and their quarrels, the troubled child who gets knocked up and then, typically, drops out of school, a jackass husband who takes the nest egg and buys beer with it. 2 thumbs up... very very very good movie.



('the', 8),  
(',', 5),  
('very', 4),  
('.', 4),  
('who', 4),  
('and', 3),  
('good', 2),  
('it', 2),  
('to', 2),  
('a', 2),  
('for', 2),  
('can', 2),  
('this', 2),  
('of', 2),  
('drama', 1),  
('although', 1),  
('appeared', 1),  
('have', 1),  
('few', 1),  
('blank', 1)  
.....

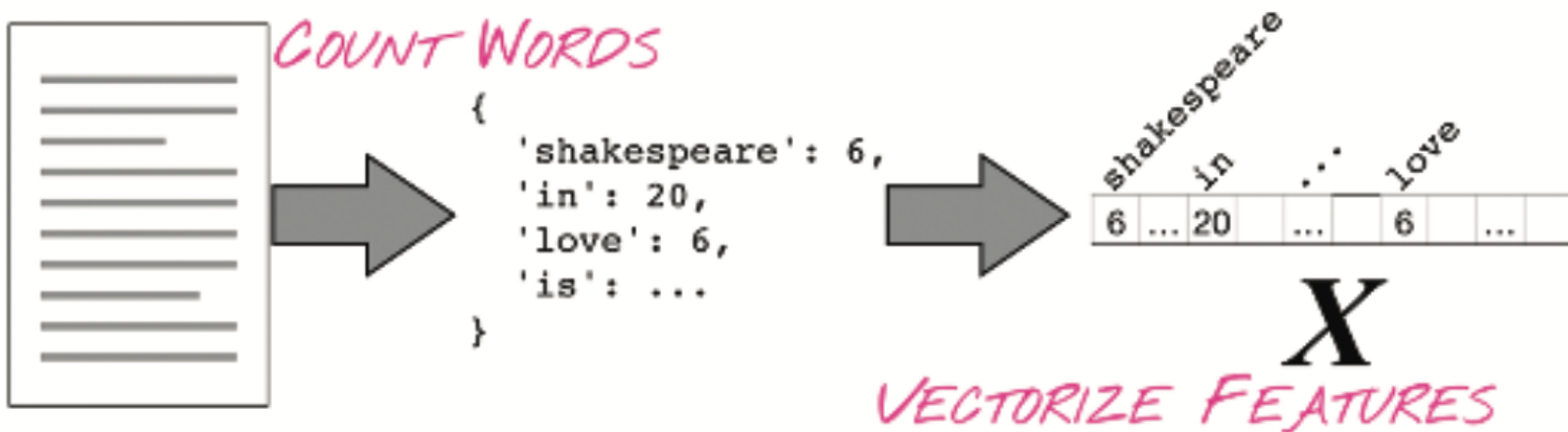


—

# Document vectors

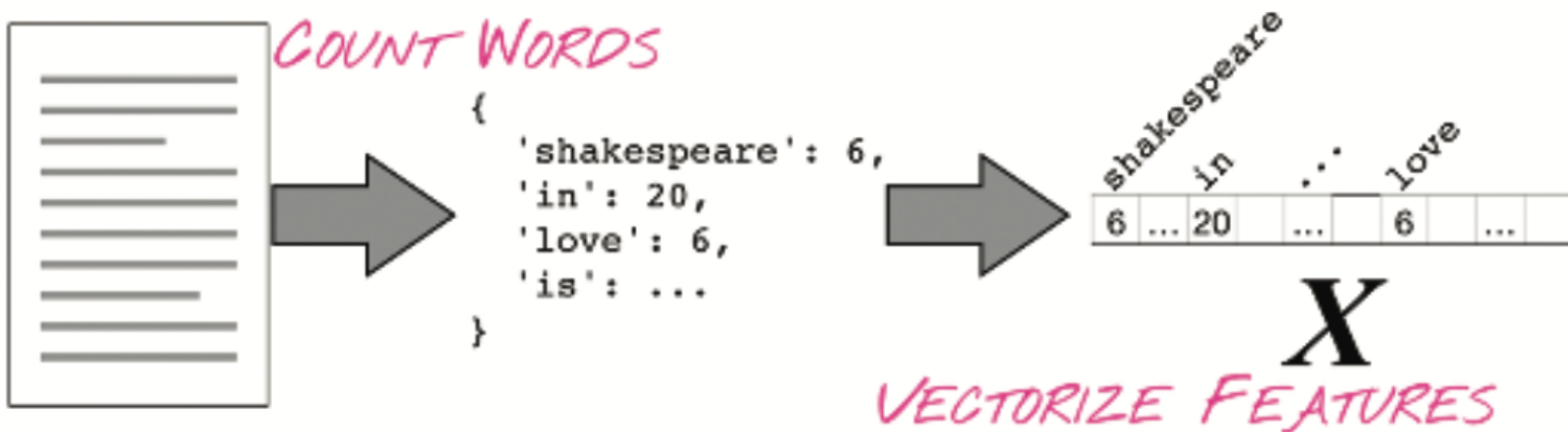
—

- Vector is just an array of numbers



- Index represents a word
- Value represents ....

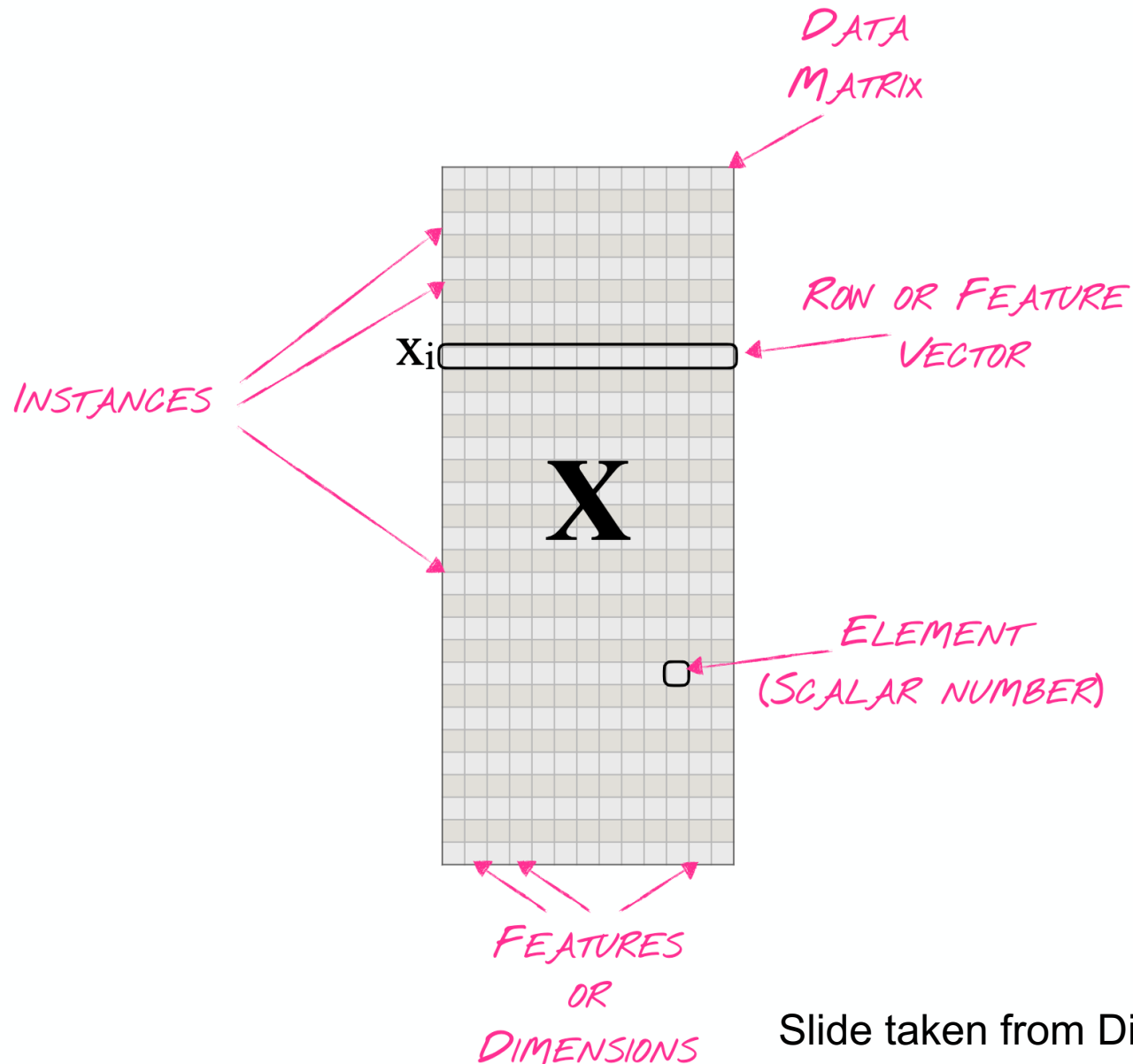
- Vector is just an array of numbers



- Index represents a word
- Value represents something about that word
  - For now word count



# Document Matrix



# Term Frequency (tf):

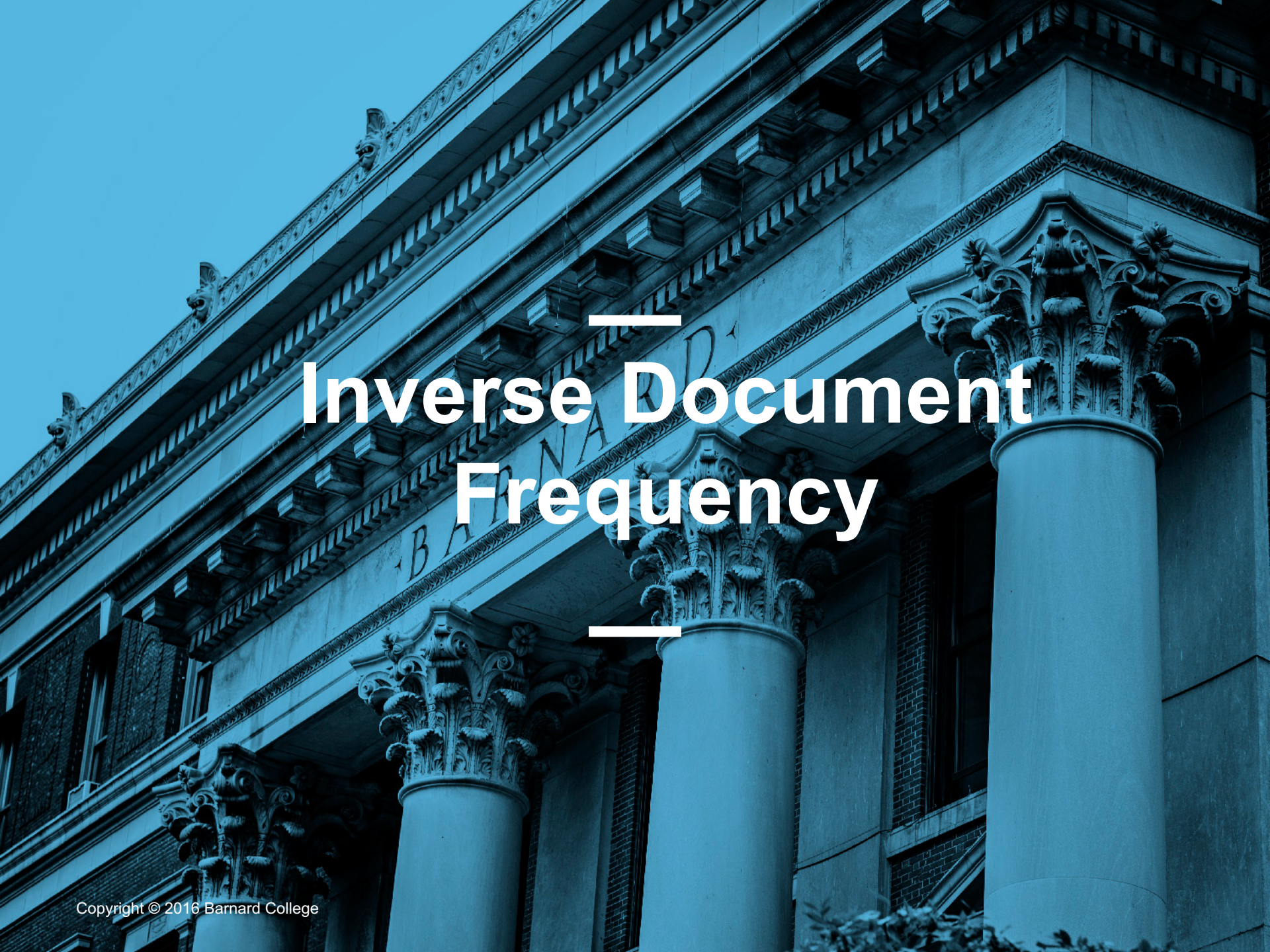


*tf* of word **w** in document **d**:

$$\frac{|w|}{| \textit{Document} |}$$

*number of times **w** appears in **D**  
divided by of number tokens in **D***

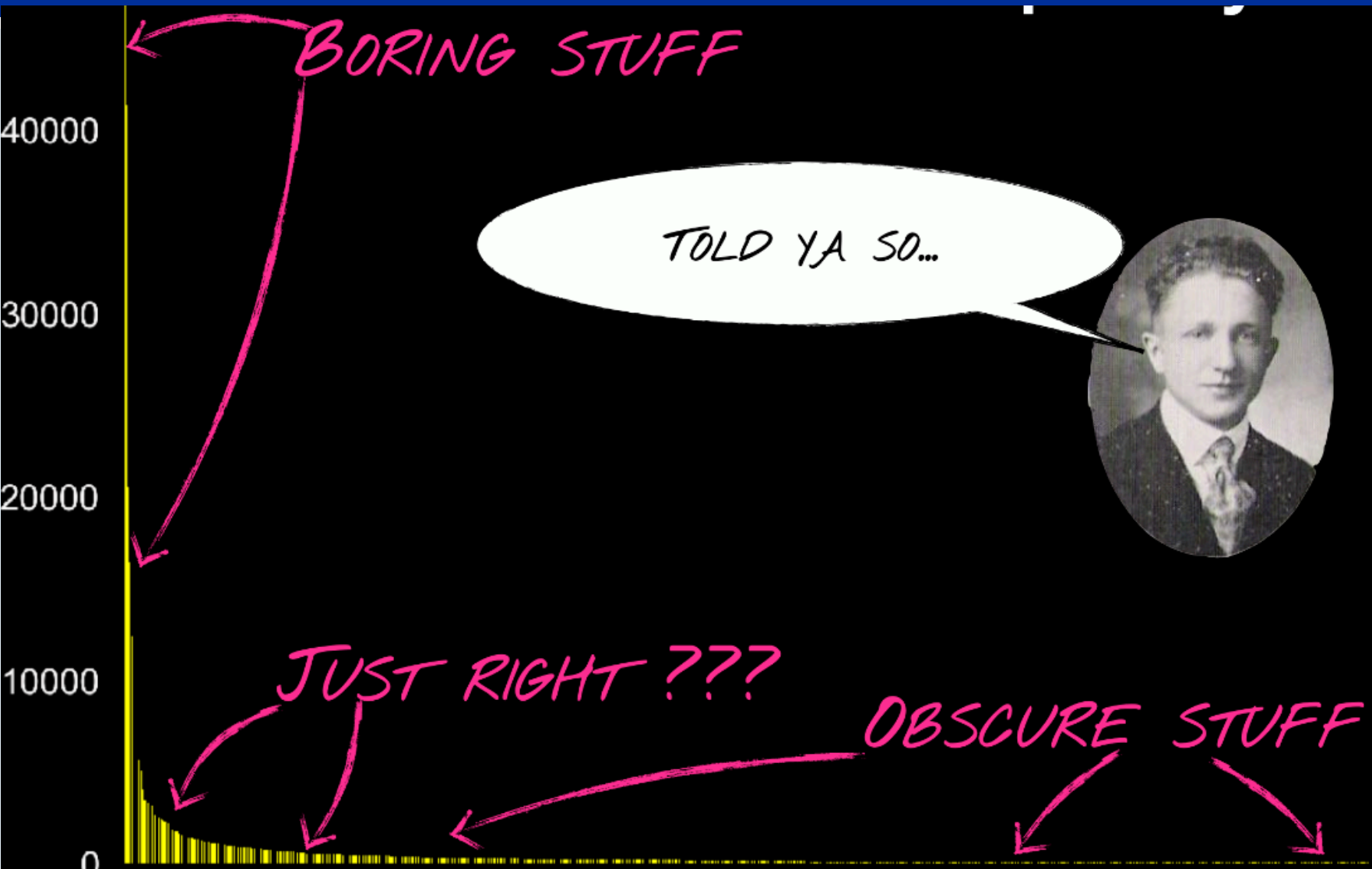




# Inverse Document Frequency



# Problem with Term Frequency





# Some words are more interesting than others



|  |   |   |  |
|--|---|---|--|
| <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> | <p>_____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> | <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> | <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p><b>sustainable</b></p> <p>_____</p> <p>_____</p> |
| <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> | <p>_____</p> <p>_____</p> <p><b>sustainable</b></p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>the _____</p>              | <p>_____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p>                           | <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>_____</p> <p>the _____</p> <p>_____</p> <p>the _____</p>      |



*idf* of word  $w$  in document  $D$ :

$$\log \frac{|D|}{|tf(w,d) \neq 0|}$$

*number of documents divided  
by number of documents that  
contain  $w$*





# TF-IDF

# TF-IDF: Term Frequency - Inverse Document Frequency



*TF-IDF* of word  $w$  in document  $D$ :

Term Frequency \* Inverse Document Frequency

Captures terms that are frequent in a document and specific to the document in the corpus





*idf* of word  $w$  in document  $D$ :

$$\log \frac{|D|}{|tf(w,d) \neq 0|}$$

*number of documents divided  
by number of documents that  
contain  $w$*