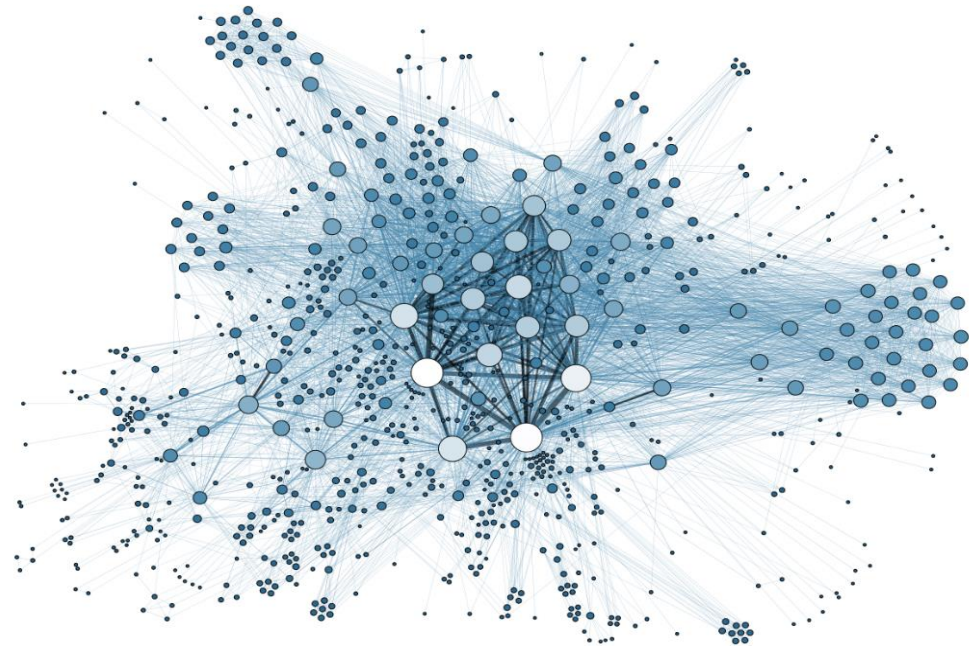




## Introduction to Text Analysis

# SOCIAL MEDIA & NETWORK ANALYTICS



# Acknowledgments

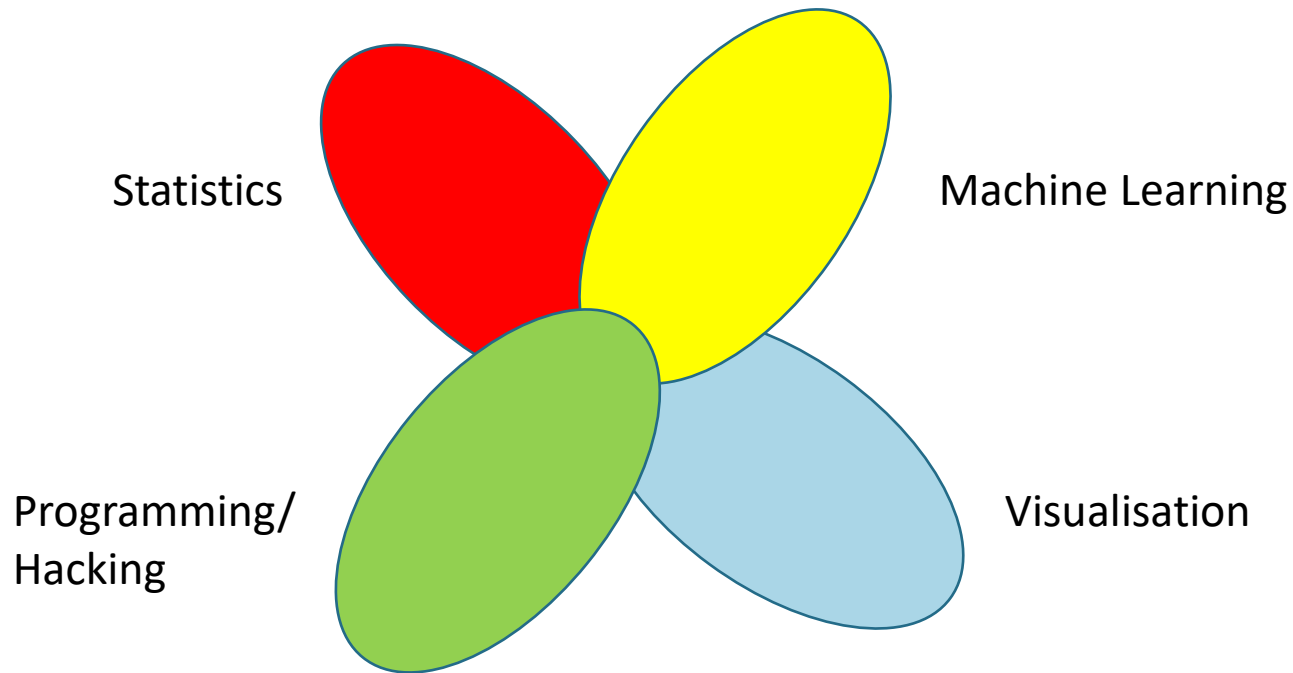
- Part of this slides based on Kai-Wei Chang slides when he was at University of Virginia

# Outline of Lecture

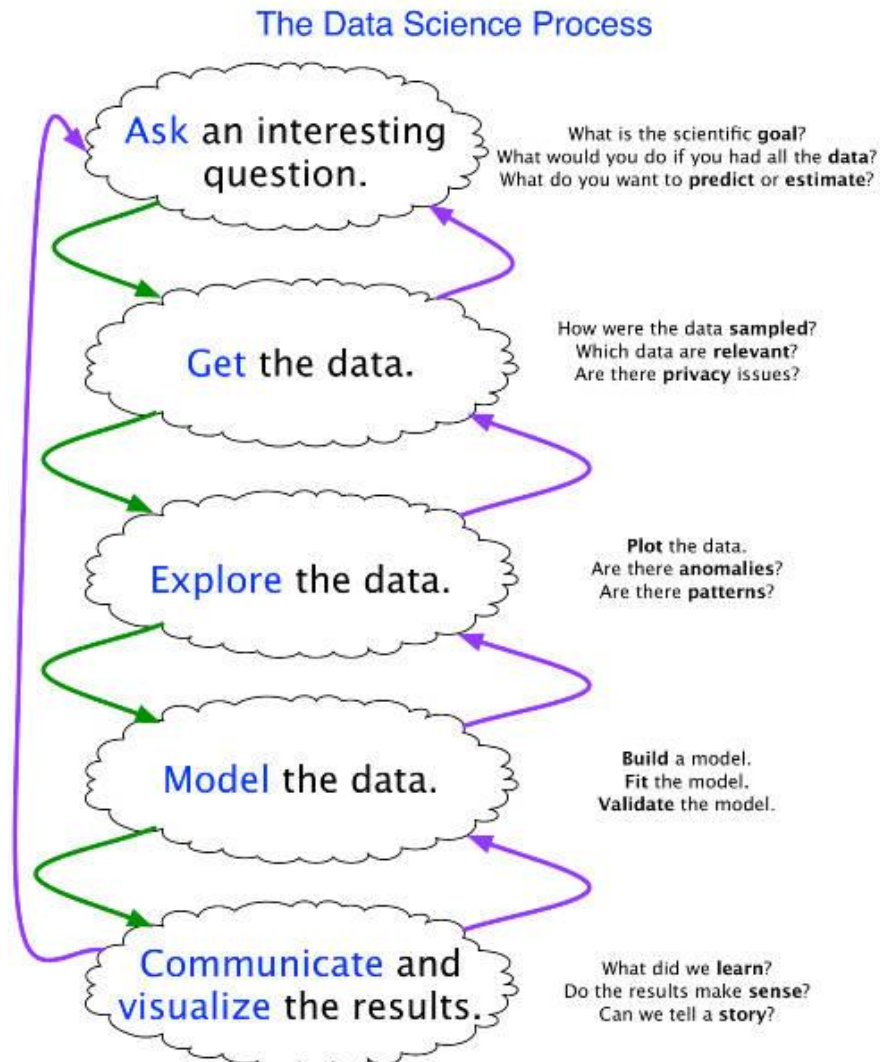
- Quick Overview of Data Science
- Introduction to Text Analysis
  - Natural Language Processing (NLP)
  - Tokenisation, Stemming/Lemmatization
  - Vector Space Models
  - Part of speech tagging
- (Text) Classification
- (Text) Clustering

# Data Science

- Analysing social media and networks is part of data science
- Data science is an intersection of machine learning, statistics, programming and visualisation/communications



# Data Science Process



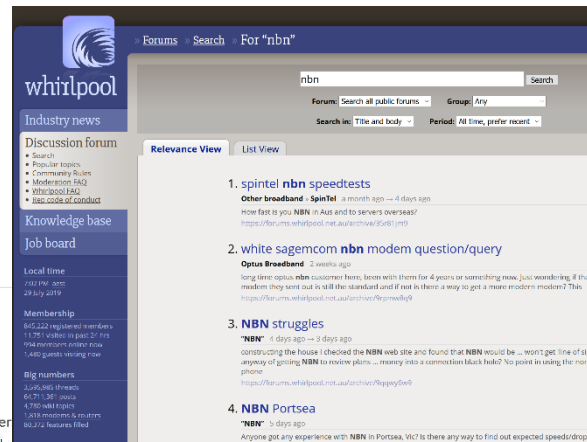
Joe Blitzstein and Hanspeter Pfister, created for the Harvard data science course <http://cs109.org/>.

# Introduction to Text Analysis

# Text – it is everywhere



## Books



## Online Forums



KRG1980  
1 review

★★★★☆ · Dined 4 days ago

Overall 4 · Food 4 · Service 5 · Ambience 3

The staff were amazingly attentive and friendly.

I had the \$69 buffet and I'm not much of a seafood consumer maybe my point of view is a bit skewed, but would have liked more variety of dishes. All the dishes were delicious. My only gripe was the main fish dish of snapper I believe it had too many bones you couldn't see because the room was so dark. On such a tiny fish, it was too tricky to deal with. I know you kind of fork off the fish and it breaks away, but quite a few still get through.

I particularly enjoyed the chicken salad, chicken and prawn dumplings and the lobster roll.

The cocktail with pineapple was also very good. Not too sweet and quite strong.

Lovely evening anyway. Cheers!

[Read less](#)

[Report](#)

## Reviews



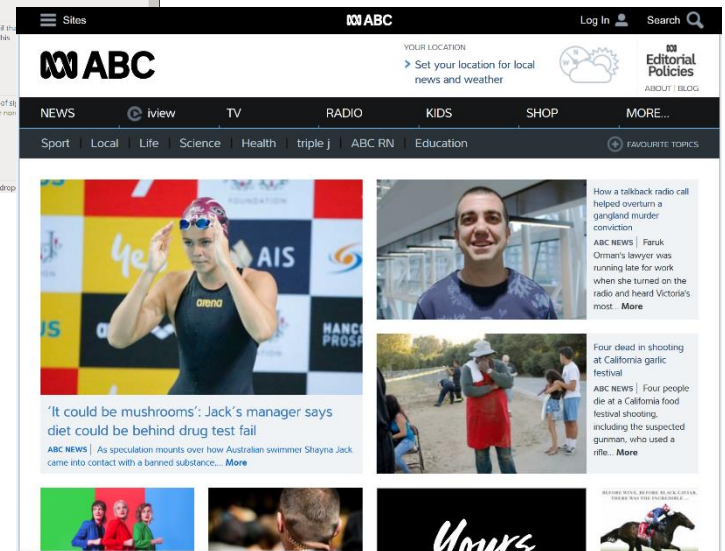
john plantus  
@jdplantus

[Follow](#)

Replying to @crispycrackling

Masterchef Australia appealed to me because of your support for the contestants (Matt's and George's, too). You three created a positive atmosphere that subverted reality TV conventions and made the world, and my life, a better place.

## Tweets



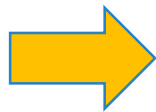
## News

# What is Text Analysis?

- Extraction of (useful) information from text
- Emphasis on automatic



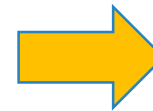
Text Extraction  
Cleaning



In case you need some revision, the labs in week 1 will be running and we'll revise some very basic Python and getting Jupyter Notebook and running. We'll also use Anaconda to install Python packages. Note this is not compulsory and if you know all this already, please no need to come. There will be no tutes running, ignore whatever the timetabling system tells you (we will instead run a Swot Vac revision tute in week 13/Swot Vac).

The computer labs that our labs are in do have Anaconda and Jupyter notebook installed. Although it isn't necessary, can I suggest it will be easier if you to come and use your laptops in labs, so you can use your setup for both labs and the assignments and avoid needing to repeat everything on your own devices when doing the assignments. Personally I found the interface on my own machine easier to use then the lab machines.

Natural Language Processing (NLP)  
Text Mining



Sentiment Analysis

Topic Detection  
(e.g., clustering)

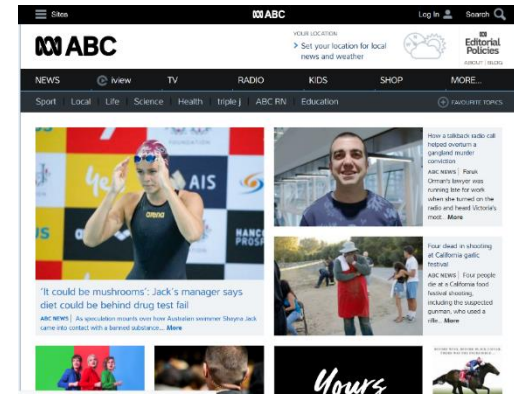
Visualisation

Classification



# Text Extraction

- **Input:** Raw Verbatim Text Data
- **Output:** Data that is clean and ready to be further processed
- **Cleaning**
  - Identifying language
  - Extract out content (e.g., news pages, have both images and text)
  - Remove boiler plate markup and text (e.g., pdf)



```
>> endobj
3 0 obj <<
  /Type /Pages
  /Kids [ 22 0 R 23 0 R ]
  /Count 2
>> endobj

22 0 obj <<
  /Type /Page
  /Parent 3 0 R
  /MediaBox [ 0 0 612 792 ]
  /Resources ...
>> endobj
```

# Natural Language Processing (NLP)

- **NLP** – a branch of AI, that develops approaches for machines to parse and understand natural language
- Essentially takes text and tries to extract meaning about words, sentences and documents.
- One of the more difficult challenges in AI.
  - *"The spirit is willing, but the flesh is weak."*
  - Translate to Russian and back
  - *"The vodka is good, but the meat is rotten."*

# NLP Tasks

- Tokenisation
  - Stemming/Lemmatisation (Morphology)
  - Frequency counts (vector space word models)
  - Part of Speech Tagging (POS)
  - Parsing Sentence Structure
- 
- Named Entity Recognition
  - Word Sense Disambiguation

# **Tokenisation**

## **Stemming/Lemmatisation**

# Tokenisation

- Splitting a sequence of characters into “words” (tokens)
- Language dependent
- *“The spirit is willing, but the flesh is weak.”*
  - [The, spirit, is, willing, but, the, flesh, is, weak]
- Approach
  - Regular expressions, e.g., words separated by whitespaces, punctuation
  - Dictionary + parser (e.g., context free grammar – CFG)
- Sometimes combined with stop-word removal
  - E.g., filter out words such as ‘the’, doesn’t convey information about sentence or document

# Stemming / Lemmatisation

- Based on Morphology
- The ways that words are built up from smaller meaningful units (**morphemes**)
- Two classes of morphemes
  - **Stems**: The core meaning-bearing units
  - **Affixes**: adhere to stems to change their meanings and grammatical functions
  - e.g.,. killing -> **kill**-**ing**, studies -> **study** (**-ies**)

# Infection Morphology

Create different forms of the same word:

- Examples:
  - Verbs: walk, walked, walks
  - Nouns: Book, books, book's
  - Personal pronouns: he, she, her, them, us
- Serves a grammatical/semantic purpose that is different from the original but is related to the original

# Stemming and Lemmatisation

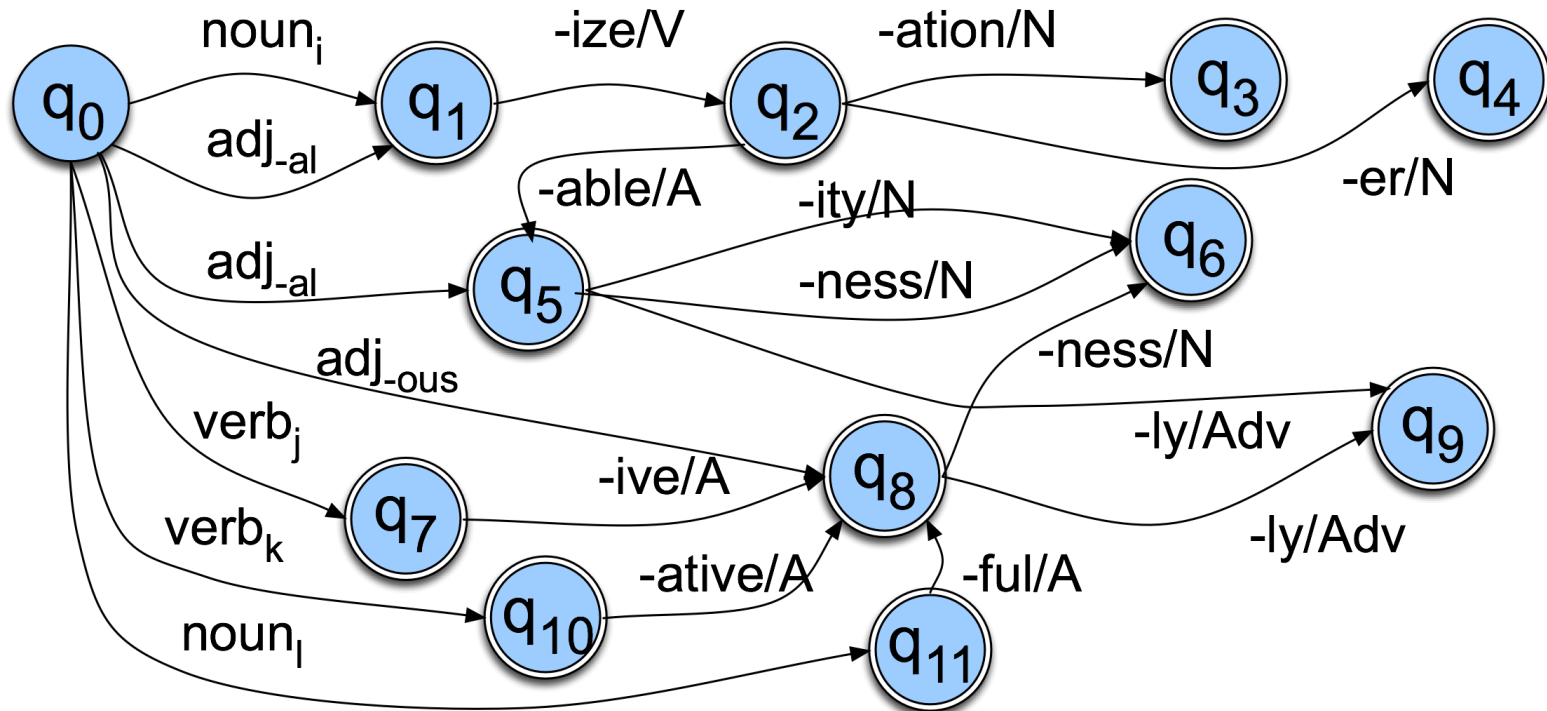
- Both try to identify the stems
- Stemming
  - The resulting stems might not be actual words
  - E.g., Studies -> Studi-es
- Approach
  - E.g., English Porter Stemmer
  - Essentially regular expressions and heuristics to remove suffixes
  - See Fig 2.8 of <https://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html>



# Lemmatisation

- Lemmatisation
  - The stems from lemmatisation are called “lemmas”
  - Lemmas are actual words in dictionary
  - E.g., Studies -> Study (ies)
- Lemmatisation Approach
  - E.g., Finite State Machines

# Finite State Machine: Derivation Rules



# Poll

- When would we use Stemming over Lemmatisation, and why would we use Lemmatisation over Stemming?
- [www.menti.com](https://www.menti.com) (enter code on screen)

# NLP Tasks

- Tokenisation
  - Stemming/Lemmatisation (Morphology)
  - Frequency counts (vector space word models)
  - Part of Speech Tagging (POS)
  - Parsing Sentence Structure
- 
- Named Entity Recognition
  - Word Sense Disambiguation

# **Vector Space (Word) Models**

# Vector Space (Word) Models

- The most common way to represent documents is to transform them into vectors
  - Process them with linear algebraic operations
- This representation is called "***Bag of Words***"
- Weights for words can be assigned by **TF-IDF**
- We can also count the frequency

# Vector Space Model

- Consider a set of documents  $D$ 
  - Each document is a set of words
- **Goal:** convert these documents to vectors

$$d_i = (w_{1,i}, w_{2,i}, \dots, w_{N,i})$$

- $d_i$  : document  $i$
- $w_{j,i}$  : the weight for word  $j$  in document  $i$

## How to set $w_{j,i}$

- Set  $w_{j,i}$  to 1 when the word  $j$  exists in document  $i$  and 0 when it does not.
- We can also set  $w_{j,i}$  to the number, of times the word  $j$  is observed in document  $i$  (frequency)

# Vector Space Model: An Example

- **Documents:**

- $d_1$ : social media mining
- $d_2$ : social media data
- $d_3$ : social financial market data

- **Dictionary of words:**

- (social, media, mining, data, financial, market)

- Vector representation:

	social	media	mining	data	financial	market
$d_1$	1	1	1	0	0	0
$d_2$	1	1	0	1	0	0
$d_3$	1	0	0	1	1	1



# TF-IDF (Term Frequency-Inverse Document Frequency)

TF-IDF of term (word)  $t$ , document  $d$ , and document corpus  $D$  is calculated as follows:

$$w_{j,i} = tf_{j,i} \times idf_j$$

$tf_{j,i}$  is the frequency of word  $j$  in document  $i$

$$idf_j = \log_2 \frac{|D|}{|\{\text{document} \in D \mid j \in \text{document}\}|}$$

The total number of documents in the corpus

The number of documents where the term  $j$  appears

# TF-IDF: An Example

Document  $d_1$  contains 100 words

- Word "apple" appears 10 times in  $d_1$
- Word "orange" appears 20 times in  $d_1$

We have  $|D| = 20$  documents

- Word "apple" only appears in document  $d_1$
- Word "orange" appears in all 20 documents

$$tf - idf(\text{"apple"}, d_1) = 10 \times \log_2 \frac{20}{1} = 43.22$$

$$tf - idf(\text{"orange"}, d_1) = 20 \times \log_2 \frac{20}{20} = 0$$

# TF-IDF: An Example

- Documents:
  - $d_1$ : social media mining
  - $d_2$ : social media data
  - $d_3$ : social financial market data

$$idf d_{social} = \log_2\left(\frac{3}{3}\right)=0$$

$$idf d_{media} = \log_2\left(\frac{3}{2}\right)=0.584$$

$$idf d_{mining} = \log_2\left(\frac{3}{1}\right)=1.584$$

$$idf d_{data} = \log_2\left(\frac{3}{2}\right)=0.584$$

$$idf d_{financial} = \log_2\left(\frac{3}{1}\right)=1.584$$

$$idf d_{market} = \log_2\left(\frac{3}{1}\right)=1.584$$

- TF values:

	social	media	mining	data	financial	market
$d_1$	1	1	1	0	0	0
$d_2$	1	1	0	1	0	0
$d_3$	1	0	0	1	1	1

- TF-IDF

	social	media	mining	data	financial	market
$d_1$	0	0.584	1.584	0	0	0
$d_2$	0	0.584	0	0.584	0	0
$d_3$	0	0	0	0.584	1.584	1.584

# N-Grams

- Previous all models count each word individually, but lose context
  - E.g., data science vs "data", "science"
- Idea of N-grams, count tuples (2-gram, adjacent pairs of words)
- "data science is interesting. big data is useful"
  - ["data science", "science is", "is interesting"...]
- What are we doing exactly?
  - Estimating probabilities:  $P(w_i, w_{i-1}, w_{i-2}, \dots) = P(w_i | w_{i-1}, w_{i-2}, \dots)$
  - Unigram:  $P(w_i, w_{i-1}, w_{i-2}, \dots) = P(w_i) P(w_{i-1}) \dots$
  - Too simple, instead:  $P(w_i, w_{i-1}, w_{i-2}, \dots) = P(w_i | w_{i-1}) P(w_{i-1} | w_{i-2}) \dots$

# NLP Tasks

- Tokenisation
  - Stemming/Lemmatisation (Morphology)
  - Frequency counts (vector space word models)
  - Part of Speech Tagging (POS)
  - Parsing Sentence Structure
- 
- Named Entity Recognition
  - Word Sense Disambiguation

# Break


- Question for break: What is the Turing test, and how is it relevant to what we just talked about?

# **Part of Speech Tagging**

# Parts of Speech (POS)

- Traditional parts of speech

## Parts of Speech



<h3>NOUN</h3> <p><i>Name of a person, place, thing or idea.</i></p> <p>Examples: Daniel, London, table, hope - <i>Mary</i> uses a blue <i>pen</i> for her <i>notes</i>.</p>	<h3>PRONOUN</h3> <p><i>A pronoun is used in place of a noun or noun phrase to avoid repetition.</i></p> <p>Examples: I, you, it, we, us, them, those - I want <i>her</i> to dance with <i>me</i>.</p>
<h3>ADJECTIVE</h3> <p><i>Describes, modifies or gives more information about a noun or pronoun.</i></p> <p>Examples: cold, happy, young, two, fun - The <i>little</i> girl has a <i>pink</i> hat.</p>	<h3>VERB</h3> <p><i>Shows an action or a state of being.</i></p> <p>Examples: go, speak, eat, live, are, is - I <i>listen</i> to the word and then <i>repeat</i> it.</p>
<h3>ADVERB</h3> <p><i>Modifies a verb, an adjective or another adverb. It tells how (often), where, when.</i></p> <p>Examples: slowly, very, always, well, too - <i>Yesterday</i>, I ate my lunch <i>quickly</i>.</p>	<h3>PREPOSITION</h3> <p><i>Shows the relationship of a noun or pronoun to another word.</i></p> <p>Examples: at, on, in, from, with, about - I left my keys <i>on</i> the table <i>for</i> you.</p>
<h3>CONJUNCTION</h3> <p><i>Joins two words, ideas, phrases together and shows how they are connected.</i></p> <p>Examples: and, or, but, because, yet, so - I was hot <i>and</i> tired <i>but</i> still finished it.</p>	<h3>INTERJECTION</h3> <p><i>A word or phrase that expresses a strong emotion. It is a short exclamation.</i></p> <p>Examples: Ouch! Hey! Oh! Watch out! - <i>Wow!</i> I passed my English exam.</p>

[www.grammar.cl](http://www.grammar.cl) [www.woodwardenglish.com](http://www.woodwardenglish.com) [www.vocabulary.cl](http://www.vocabulary.cl)



# POS Tagging

- The process of assigning a part-of-speech to each word in a collection (sentence).

Jeff	is	an	academic
Noun	verb	det.	noun

# Penn TreeBank POS Tagset

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	<i>and, but, or</i>	SYM	symbol	<i>+, %, &amp;</i>
CD	cardinal number	<i>one, two, three</i>	TO	“to”	<i>to</i>
DT	determiner	<i>a, the</i>	UH	interjection	<i>ah, oops</i>
EX	existential ‘there’	<i>there</i>	VB	verb, base form	<i>eat</i>
FW	foreign word	<i>mea culpa</i>	VBD	verb, past tense	<i>ate</i>
IN	preposition/sub-conj	<i>of, in, by</i>	VBG	verb, gerund	<i>eating</i>
JJ	adjective	<i>yellow</i>	VCN	verb, past participle	<i>eaten</i>
JJR	adj., comparative	<i>bigger</i>	VBP	verb, non-3sg pres	<i>eat</i>
JJS	adj., superlative	<i>wildest</i>	VBZ	verb, 3sg pres	<i>eats</i>
LS	list item marker	<i>1, 2, One</i>	WDT	wh-determiner	<i>which, that</i>
MD	modal	<i>can, should</i>	WP	wh-pronoun	<i>what, who</i>
NN	noun, sing. or mass	<i>llama</i>	WP\$	possessive wh-	<i>whose</i>
NNS	noun, plural	<i>llamas</i>	WRB	wh-adverb	<i>how, where</i>
NNP	proper noun, singular	<i>IBM</i>	\$	dollar sign	<i>\$</i>
NNPS	proper noun, plural	<i>Carolinas</i>	#	pound sign	<i>#</i>
PDT	predeterminer	<i>all, both</i>	“	left quote	<i>‘ or “</i>
POS	possessive ending	<i>’s</i>	”	right quote	<i>’ or ”</i>
PRP	personal pronoun	<i>I, you, he</i>	(	left parenthesis	<i>[, (, {, &lt;</i>
PRP\$	possessive pronoun	<i>your, one’s</i>	)	right parenthesis	<i>], ), }, &gt;</i>
RB	adverb	<i>quickly, never</i>	,	comma	<i>,</i>
RBR	adverb, comparative	<i>faster</i>	.	sentence-final punc	<i>. ! ?</i>
RBS	adverb, superlative	<i>fastest</i>	:	mid-sentence punc	<i>: ; ... --</i>
RP	particle	<i>up, off</i>			

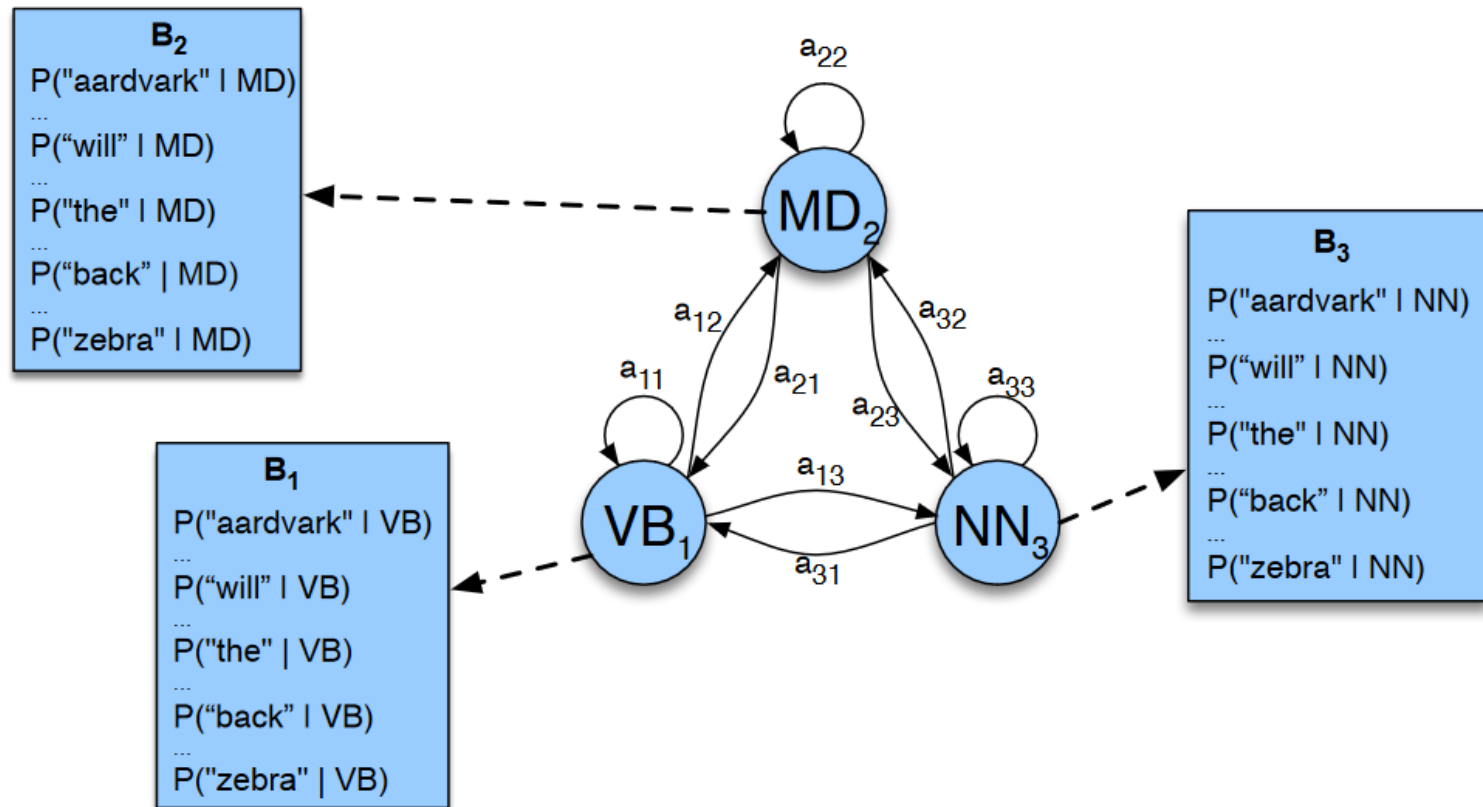
# Why is POS Tagging Useful?

- First step of a vast number of practical tasks
- Parsing
  - Need to know if a word is an N or V before you can parse
- Information extraction
  - Finding names, relations, etc.
- Speech synthesis/recognition
- Machine Translation

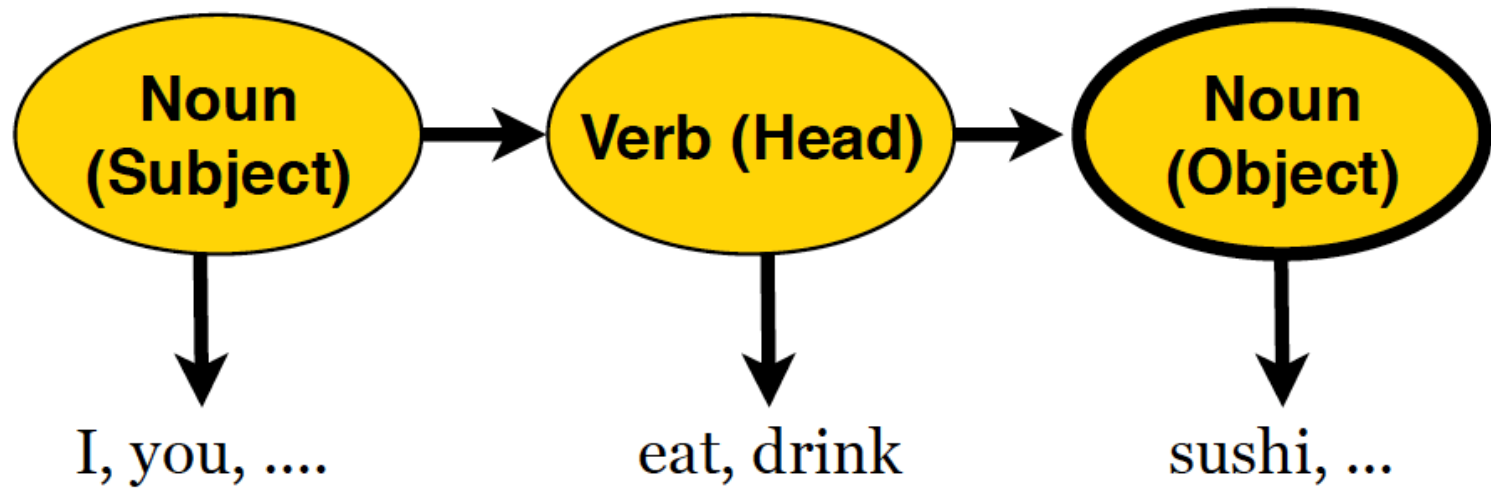
# POS Tagging Approaches

- Rule-based Taggers
  - If X and Y and Z, then its POS A
  - Difficult to enumerate all possible rules
  - E.g., Brill tagger, is a rule based miner that discovers relevant rules from training data based on rule templates
- Statistical Taggers
  - Use both context (what tags occur before and after) as well as frequency of a word being of certain tag
  - E.g., Hidden Markov Models, transitions between states represent the tag context, while emission probabilities the frequency

# HMM Tagger



# Example



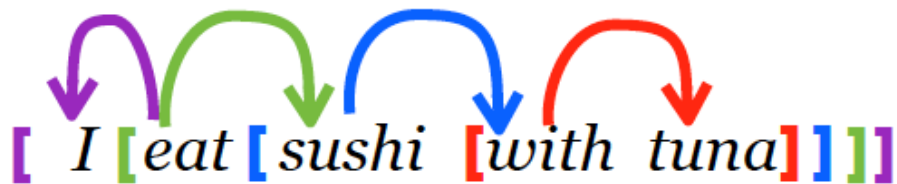
- I eat sushi; I eat meat; you eat banana...

# Hierarchical Parsing

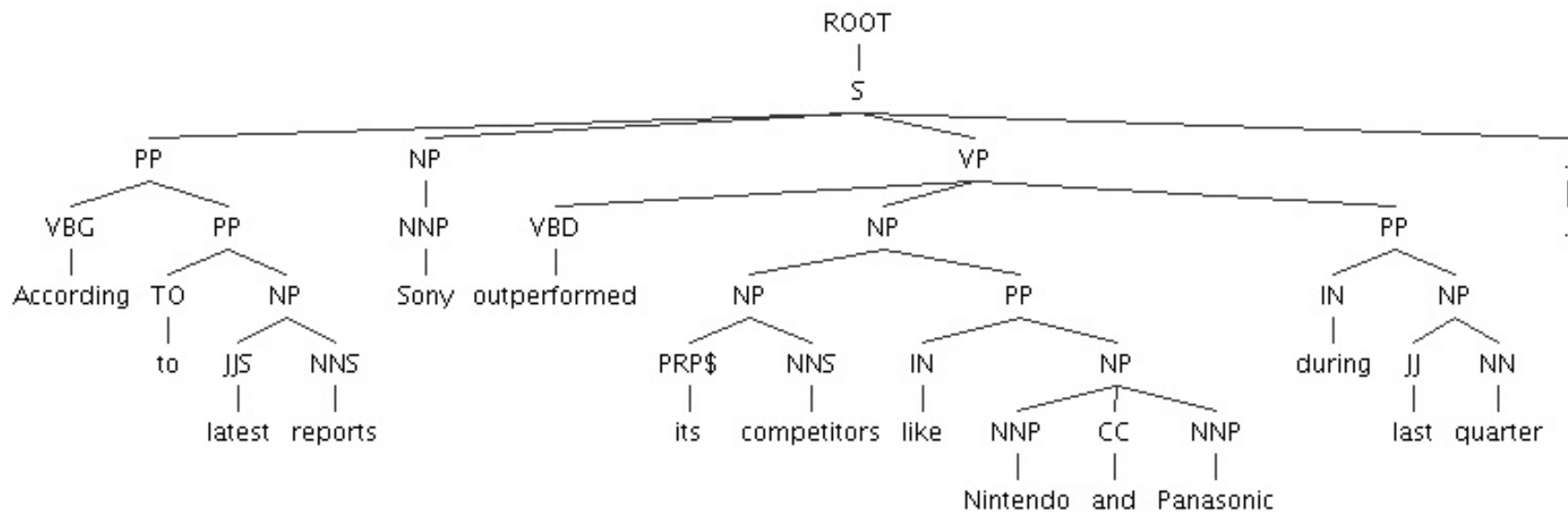
- So far, all the models we saw formulate sentence as a sequence (flat)
- Sentence structure is often hierarchical
  - A sentence consists of phrases (or constituents)

.

Sentence structure defines dependencies between words or phrases:



# Hierarchical POS Tagging



**According to latest reports Sony outperformed its competitors like Nintendo and Panasonic during last quarter.**



# NLP Tasks

- Tokenisation
  - Stemming/Lemmatisation (Morphology)
  - Frequency counts (vector space word models)
  - Part of Speech Tagging (POS)
  - Parsing Sentence Structure
- 
- Named Entity Recognition
  - Word Sense Disambiguation

# **Text Classification and Clustering (Machine Learning)**

# Text Classification and Clustering

- Text Classification (Supervised learning)
  - Classification approaches
  - Evaluation
- Text Clustering (Unsupervised learning)
  - Clustering approaches
  - Evaluation

# Text Classification

- Set of blogs/tweets/documents with known topic labels



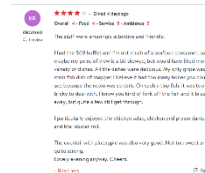
Food



Food



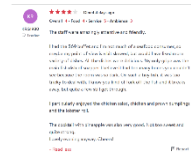
Travel



Travel



Food



Travel

- Given new blog/tweet, want to classify into one of the known topics



Food or Travel?

# Approach

Data Processing



Processed (text) data

Classification



Classification output

Evaluation

# Data Pre-processing

- **Outliers**

- Outliers are data points that are considerably different from other data points in the dataset

- **Missing Values**

- Missing feature values in data instances

- **Solution:**

- Remove instances that have missing values
- Estimate missing values, and
- Ignore missing values when running machine learning algorithm

- **Remove Duplicate data**

# Data Pre-processing

- **Aggregation**

- It is performed when multiple features need to be combined into a single one or when the scale of the features change
- Example: image width , image height -> image area (width x height)

- **Discretization**

- From continuous values to discrete values
- Example: money spent -> {low, normal, high}

- **Feature Selection**

- Choose relevant features

- **Feature Extraction**

- Creating new features from original features
- Often, more complicated than aggregation

- **Sampling**

- Random Sampling
- Sampling with or without replacement
- Stratified Sampling: useful when having class imbalance
- Social Network Sampling

# Approach

Data Processing



Processed (text) data

Classification



Classification output

Evaluation



# Supervised Learning Algorithms

- **Classification**

- Decision tree learning
- Naive Bayes Classifier
- $k$ -nearest neighbour classifier
- Support vector machines

- **Regression**

- Linear Regression

# Approach

Data Processing



Processed (text) data

Classification



Classification output

Evaluation

# Evaluating Supervised Learning

- Training/Testing Framework:
  - A training dataset (i.e., the labels are known) is used to train a model
  - the model is evaluated on a test dataset.
- When testing, the labels from this test set are removed.
  - After these labels are predicted using the model, the predicted labels are compared with the masked labels (**ground truth**).

# Evaluating Supervised Learning

- Dividing the training set into train/test sets
  - **Leave-one-out training**
    - Divide the training set into  $k$  equally sized partitions
      - Often called **folds**
    - Use all folds but one to train and the one left out for testing
  - **$k$ -fold cross validation training**
    - Divide the training set into  $k$  equally sized sets
    - Run the algorithm  $k$  times
    - In round  $i$ , we use all folds but fold  $i$  for training and fold  $i$  for testing.
    - The average performance of the algorithm over  $k$  rounds measures the performance of the algorithm.

# Evaluating Supervised Learning

- As the class labels are discrete, we can measure the accuracy by dividing number of correctly predicted labels ( $C$ ) by the total number of instances ( $N$ )

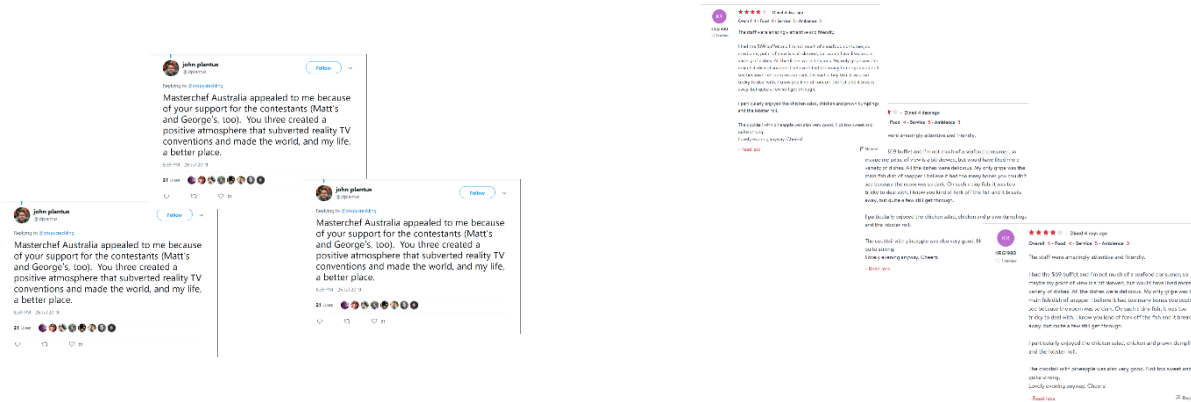
$$\text{accuracy} = \frac{C}{N}$$

$$\text{error rate} = 1 - \text{accuracy}$$

- More sophisticated approaches of evaluation
  - AUC
  - F-Measure

# Text Clustering

- Set of documents/tweets, but this time all unknown labels



- Want to find topics groups?
- Assume there are  $k$  number of topics

# Text Clustering Approach

Data Processing



Processed (text) data

Clustering



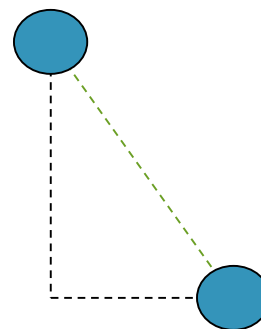
Clustering output

Evaluation

# Measuring Distance/Similarity in Clustering

- **Clustering Goal:** Group together similar items
- Instances are put into different clusters based on the distance to other instances
- **Any clustering algorithm requires a distance measure**

The most popular (dis)similarity measure for continuous features are **Euclidean Distance** and **Pearson Linear Correlation**



Euclidean Distance

$$d(X, Y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$



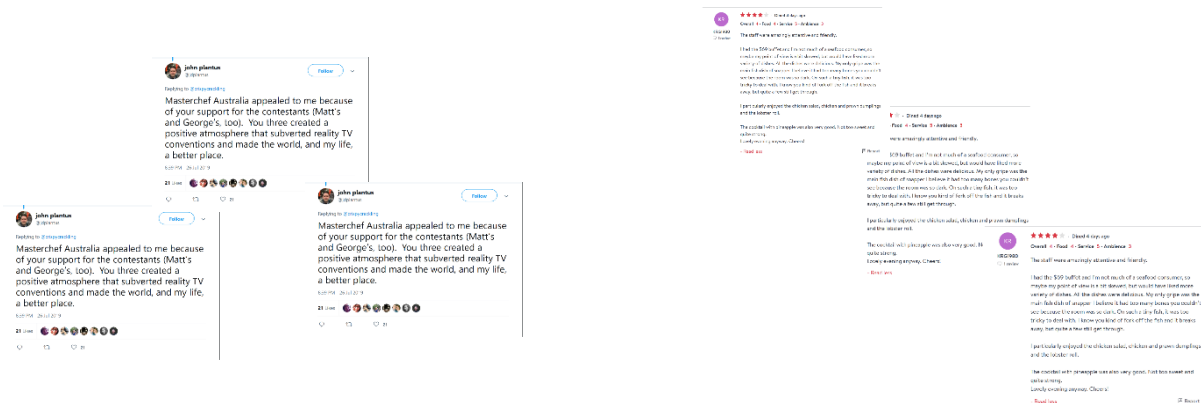
# Clustering Approaches

- Clustering

- K-Means

- DBScan

- Hierarchical



# Text Clustering Approach

Data Processing



Processed (text) data

Clustering



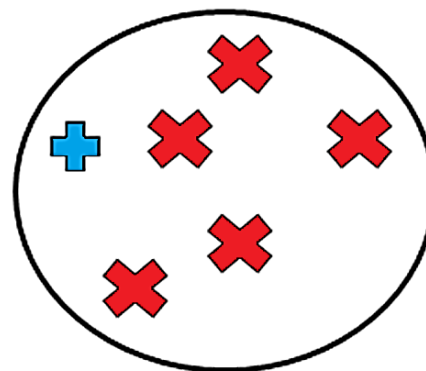
Clustering output

Evaluation

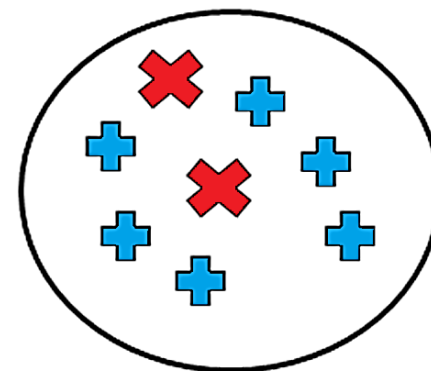
# Evaluating the Clusterings

We are **given** two types of objects

- In **perfect clustering**, objects of the same type are clustered together.



*Cluster 1*



*Cluster 2*

- Evaluation **with ground truth**
- Evaluation **without ground truth**

# Evaluation with Ground Truth

When ground truth is available,

- We have prior knowledge on what the clustering should be (the correct clustering assignments)
- We will discuss more about these methods in community analysis lecture

# Evaluation without Ground Truth

- **Cohesiveness**

- In clustering, we are interested in clusters that exhibit cohesiveness
- In cohesive clusters, instances inside the clusters are close to each other

- **Separateness**

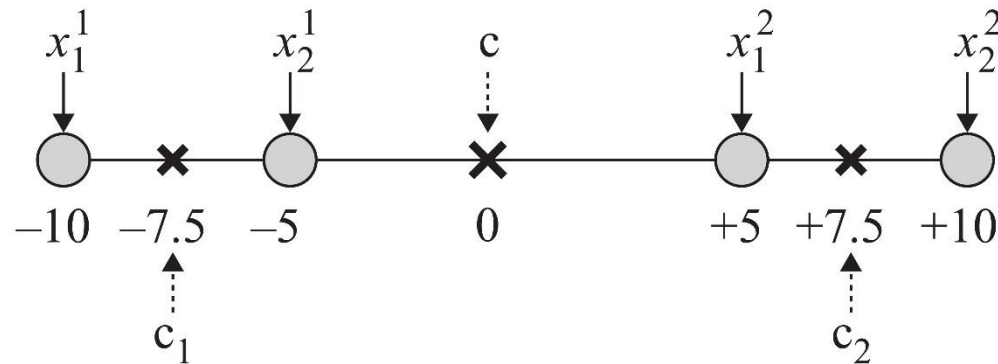
- We are also interested in clusterings of the data that generates clusters that are well separated from one another

# Cohesiveness

- **Cohesiveness**

- **In statistics:** having a small standard deviation, i.e., being close to the mean value
- **In clustering:** being close to the centroid of the cluster

$$cohesiveness = \sum_{i=1}^k \sum_{j=1}^{n(i)} ||x_j^i - c_i||^2$$



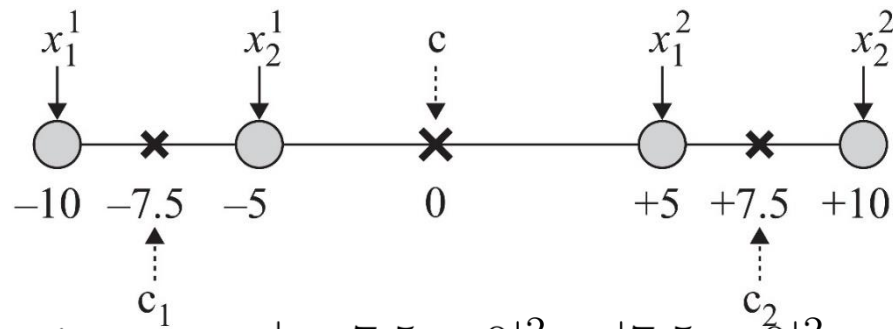
$$cohesiveness = |-10 - (-7.5)|^2 + |-5 - (-7.5)|^2 + |5 - 7.5|^2 + |10 - 7.5|^2 = 25$$

# Separateness

- **Separateness**

- **In statistics:** separateness can be measured by standard deviation
  - Standard deviation is maximized when instances are far from the mean
- **In clustering:** cluster centroids being far from the mean of the entire dataset

$$separateness = \sum_{i=1}^k ||c - c_i||^2$$



$$separateness = |-7.5 - 0|^2 + |7.5 - 0|^2 = 112.5$$

# Summary

- Introduced text analysis
- Text pre-processing
- NLP
  - Tokenisation, Stemming, POS Tagging
  - Vector space models, TF-IDF
- (Text) Classification
- (Text) Clustering



# Todo

- Revise the machine learning concepts if you not familiar
- Do quiz