Language

NLP: Semantic Similarity

Katja Markert
School of Computing
University of Leeds
markert@comp.leeds.ac.uk

Previous lecture

- Corpora
- BoW models for whole texts and queries and texts
- **This lecture: models of word meaning.**

Overview

- Models of meaning: what can we do with just words
- Semantic similarity
- Vector space model
- Vector space measures
- Clustering (revision from second year)

Semantic Similarity

Suppose you are given the following words. Your task is to group them according to how similar they are:

apple	apple
banana	banana
man	grapefruit
grapefruit	watermelon
woman	grape
baby	man
watermelon	woman
infant	baby
grape	infant

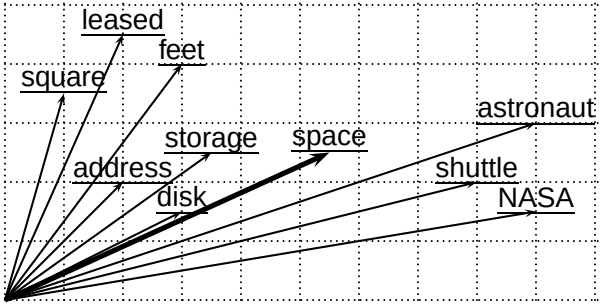
Do this automatically!
By processing text!

Vector Space Model

You shall know a word by the company it keeps (Firth,57).

- Word meaning $\approx$ context.
- Words are similar if they occur in similar contexts.
- Words are represented as vectors in multidimensional space.
- Words with vectors closer in space are more similar.
- Apart from the Distributional Measures treated here, there are also Thesaurus-based Methods (cf. JM chapter 20.6)

Vector Space Model



- Is **space** closer to **NASA** or **square**?
- What is the word most similar to **space**?
- But how do we construct these vectors?

Vector Space Model

- Each word is represented as a vector.
- The components of the vector are labelled with other words (**context words** or **features**).
- Value of each component is an association measure of target word with component label.
- Simplest: within a predefined **window** of words.
- Co-occurrence vector of word $\approx$ high dimensional summary of its behaviour.

Example: Vector Space Model

And this is the last thing the anthropologist who tries to understand symbols can afford to do. he must start with the explanations and commentaries which his informants themselves offer about their symbols. these must **first** be examined in the contexts in which they are usually employed, where they occur naturally, although subsequent generalizing discussion helps the anthropologist to improve his very superficial initial standing. **learning** the meaning of symbols is part of the anthropologist's practical semantics: **discovering** the meaning of words, noticing when their use is appropriate and when it is not. all this requires imagination, patience, considerable linguistic skill, but above all a rigorous respect for the facts. these must come **first**; fantasy can come later ...

	these	meaning	the	practical	come
first	2	0	0	0	2
learning					
discovering					

Parameters

Steps in construction of vector space models:

1. First determine a set of **context words**, e.g., the 500 most frequent words in the corpus.
2. Then determine a **window size**, e.g., 10 words (5 to the left and 5 to the right).
3. For each word in the corpus, compute a **co-occurrence vector** using an association measure, e.g., how often the word co-occurs with the context words in the given window.
4. Use a **similarity** measure to compute whether vectors are close in space.

Variations: Context Features

As replacement or refinement of words:

- Eliminate stop words (function words) from the context features. Why?
- Words vs. lemmas or stems
- Use (grammatical function, word) pairs instead of words alone. Why?
  - How often does *student* occur as subject of *learn*, how often as *object*?
  - How often does *meaning* occur as subject of *learn*, how often as *object*?

Variations: Size of context

Nearest Neigbours of car and dog (BNC)
2-word window

car	dog
van	cat
vehivle	fox
truck	fox
motorcyle	pet
driver	rabbit
motor	pig

tendency: paradigmatic associations

Variations: Size of context

30-word window:

car	dog
drive	kennel
park	puppy
bonnet	pet
windscreen	bitch
headlight	rottweiler

Tendency: syntagmatic associations

Variations: Use of association measure

- Binary co-occurrence: 1, 0
- Co-occurrence frequency (why?)
- Pointwise mutual information (why?)

$$PMI(f_i, w) = log \frac{p(f_i, w)}{p(w) \times p(f_i)}$$

Objects of *drink* (Hindle, 1990)

object	co-occ count	PMI
tea	4	11.75
Pepsi	2	11.75
beer	5	10.20
much	3	2.54
it	3	1.25

Variations: Similarity Measures

Each word is represented as a vector of n values:

$$\vec{x} = (x_1, x_2, \ldots, x_n)$$

The **cosine** of the angle between two vectors $\vec{x}$ and $\vec{y}$ is:

$$cos(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{|\vec{x}||\vec{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2} \sqrt{\sum_{i=1}^{n} y_i^2}}$$

The **Euclidian** distance of two vectors $\vec{x}$ and $\vec{y}$ is:

$$|\vec{x} - \vec{y}| = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Cosine is more frequently used than Euclidean distance. Why?

Example: Similarity measures

	cosmonaut	astronaut	moon	car	truck
cosmonaut	**2**	**0**	**1**	**1**	**0**
astronaut	0	1	1	0	0
moon	**1**	**1**	**2**	**1**	**0**
car	1	0	1	3	1
truck	0	0	0	1	2

$$cos(\vec{cosmonaut}, \vec{moon}) = \frac{2 \cdot 1 + 0 \cdot 1 + 1 \cdot 2 + 1 \cdot 1 + 0 \cdot 0}{\sqrt{2^2 + 0^2 + 1^2 + 1^2 + 0^2} \sqrt{1^2 + 1^2 + 2^2 + 1^2 + 0^2}} = \frac{5}{\sqrt{6}\sqrt{7}} = \frac{5}{6.481} = 0.77$$

Example: Similarity Measures

	cosmonaut	astronaut	moon	car	truck
cosmonaut	**2**	**0**	**1**	**1**	**0**
astronaut	0	1	1	0	0
moon	1	1	2	1	0
car	1	0	1	3	1
truck	**0**	**0**	**0**	**1**	**2**

$$cos(\vec{cosmonaut}, \vec{truck}) = ?$$

Evaluation

- Intrinsic: Human association norms such as Rubenstein and Goodenough (1965): 65 word pairs scored on a scale of 0-4
- In an application (extrinsically)
  - Automatic Thesaurus extraction and expansion (Grefenstette 1994, Lin 1998, Pantel 2000, Rapp 2004)
  - Detection of malapropism (contextual misspelling): “It is minus 15, and then there is the windscreen factor on top of that “ (Jones and Martin 1997)
  - Word Sense disambiguation
  - Synonym tasks and other language tests (Landauer and Dumais 1997; Turney et al 2003): see Toefl test. Best VSM 64.5%; real applicants 64.5%, native speakers 97.75% (Rapp 2004)
  - Query expansion in information retrieval

Clustering

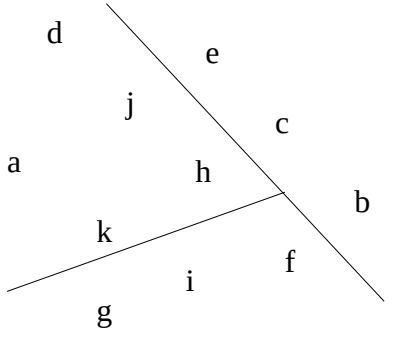
- **Clustering:** learn a classification from the data.
- Clustering algorithms divide a data set into **natural groups** (clusters). Instances in the same cluster are **similar** to each other, they share certain properties.
- **Unsupervised learning:** clustering is an unsupervised task, the training data doesn’t specify what we are trying to learn (the clusters).
- **Exploratory data analysis:** visualize the data at hand, get a feeling for what the data look like, what its properties are. First step in building a model.

Clustering Algorithms

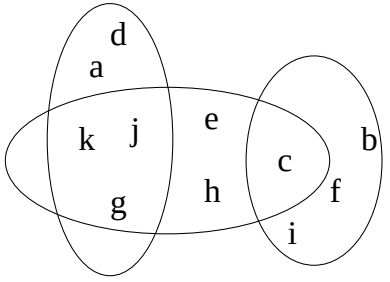
Clustering algorithms can have different properties:

- **Hierarchical or flat:** hierarchical algorithms induce a hierarchy of clusters of decreasing generality, for flat algorithms, all clusters are the same.
- **Hard and soft:** hard clustering assigns each instance to exactly one cluster. Soft clustering assigns each instance a probability of belonging to a cluster.
- **Disjunctive:** instances can be part of more than one cluster.
- **Iterative:** the algorithm starts with initial set of clusters and improves them by reassigning instances to clusters.

Examples

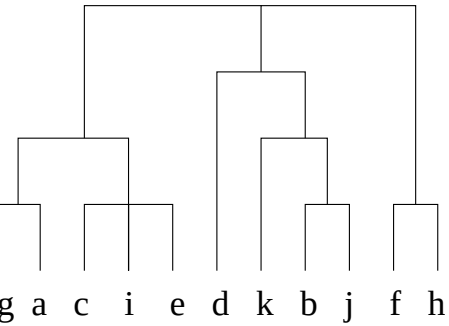


Hard, non-hierarchical, non-disjunctive



Hard, non-hierarchical, disjunctive

Examples



Hard, hierarchical, non-disjunctive

	1	2	3
a	0.4	0.1	0.5
b	0.1	0.8	0.1
c	0.3	0.3	0.4
d	0.1	0.1	0.8
e	0.4	0.2	0.4
f	0.1	0.4	0.5
g	0.7	0.2	0.1
h	0.5	0.4	0.1

Soft, non-hierarchical, disjunctive

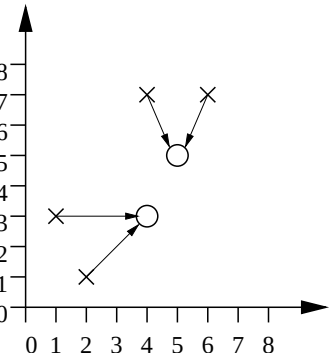
The k-means Algorithm

Iterative, hard, flat clustering algorithm based on Euclidian distance. Intuitive formulation:

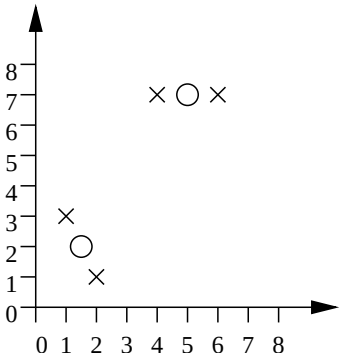
- Specify k, the number of clusters to be generated.
- Choose k points at random as cluster centers.
- Assign each instance to its closest cluster center using Euclidian distance.
- Calculate the centroid (mean) for each cluster, use it as new cluster center.
- Reassign all instances to the closest cluster center.
- Iterate until the cluster centers don't change any more.

The k-means Algorithm

Example:



Assign instances (crosses) to closest cluster centers (circles) according to Euclidian distance



Recompute cluster centers as means of the instances in each cluster

http://home.dei.polimi.it/matteucc/Clustering/tutorial_htm

The k-means Algorithm

Properties of the algorithm:

- It only finds a **local maximum**, not a global one.
- The clusters it comes up with depend a lot on which **random cluster centers** are chose initially.
- Can be used for **hierarchical clustering**: first apply k-means with $k = 2$, yielding two clusters. Then apply it again on each of the two clusters, etc.
- **Distance metrics** other than Euclidian distance can be used, e.g., the cosine.

Evaluating Clustering Models

Set the model parameters on the training set. Then test its performance on the test set, keeping the parameters constant.

Training set: set of instances used to generate the clusters (compute the cluster centers in the case of k-means).

Test set: a set of unseen instances classified using to the clusters generated during training.

Problem: How do we determine k, i.e., the number of clusters? (Many algorithms require a fixed k, not only k-means.)

Solution: treat k as a parameter, i.e., vary k and find the best performance on a given data set.

Evaluating Clustering Models

Problem: How do we evaluate the performance on the test set? How do we know if the clusters are correct?

Summary

- Vector space model
- Semantic similarity
- Similarity measures
  - cosine
  - Euclidian distance
  - many more $\cdots$
- Clustering
  - k-means

Reading: Chapter 20.7 in J&S