CIS 530: Vector Semantics

JURAFSKY AND MARTIN CHAPTER 6

Reminders

- Quiz 2 on n-gram LMs is due tonight before 11:59pm.
- Homework 3 is due on Wednesday
- Read Textbook Chapters 3 and 6

Word Meaning

How should we **represent** the **meaning** of a word?

In N-gram LMs we represented words as a string of letters or as an index in a vocabulary list.

Ideally, we want a meaning representation to encode:

1. **Synonyms** – words that have similar meanings
2. **Antonyms** – words that have opposite meanings
3. **Connotations** – words that are positive or negative
4. **Semantic Roles** – *buy, sell*, and *pay* are different parts of the same underlying *purchasing* event
5. Support for **inference**

Dictionary Definitions

Noun

1. A small insect.
2. A harmful microorganism, as a bacterium or virus.
3. An enthusiastic, almost obsessive, interest in something.
   ‘they caught the sailing bug’
4. A miniature microphone, typically concealed in a room or telephone, used for surveillance.
5. An error in a computer program or system.

Verb

1. Conceal a miniature microphone in (a room or telephone) in order to monitor or record someone's conversations.
2. Annoy or bother (someone)

Polysemy

A lemma that has multiple meanings is called **polysemous**. We call each of these aspects of the meaning of *bug* a **word sense**.

Polysemy can make interpretation difficult.

What if someone types “caught a bug” into Google?

Word sense disambiguation is the task of determining which sense of a word is being used in a context.

Synonymy

When one word has a sense whose meaning is nearly identical to a sense of another word then those two words are **synonyms**.

glitch/error

microbe/bacterium

insect/pest

microphone/wire

Formally, two words are synonymous if they are **substitutable** one for the other in **any sentence** without changing the **truth conditions** of the sentence.

In logic, that means the two words carry the same **propositional meaning**.

Principle of Contrast

Linguists assume that **a difference in form** is always associated with a **difference in meaning**.

While substitutions like *water/H_2O* or *father/dad* are truth preserving, the words are still not identical in meaning.

H_2O is used in scientific contexts, but not general texts like hiking guides

Father is a more formal version of *dad.*

It is possible that no two words have **absolutely identical** meaning.

Word similarity

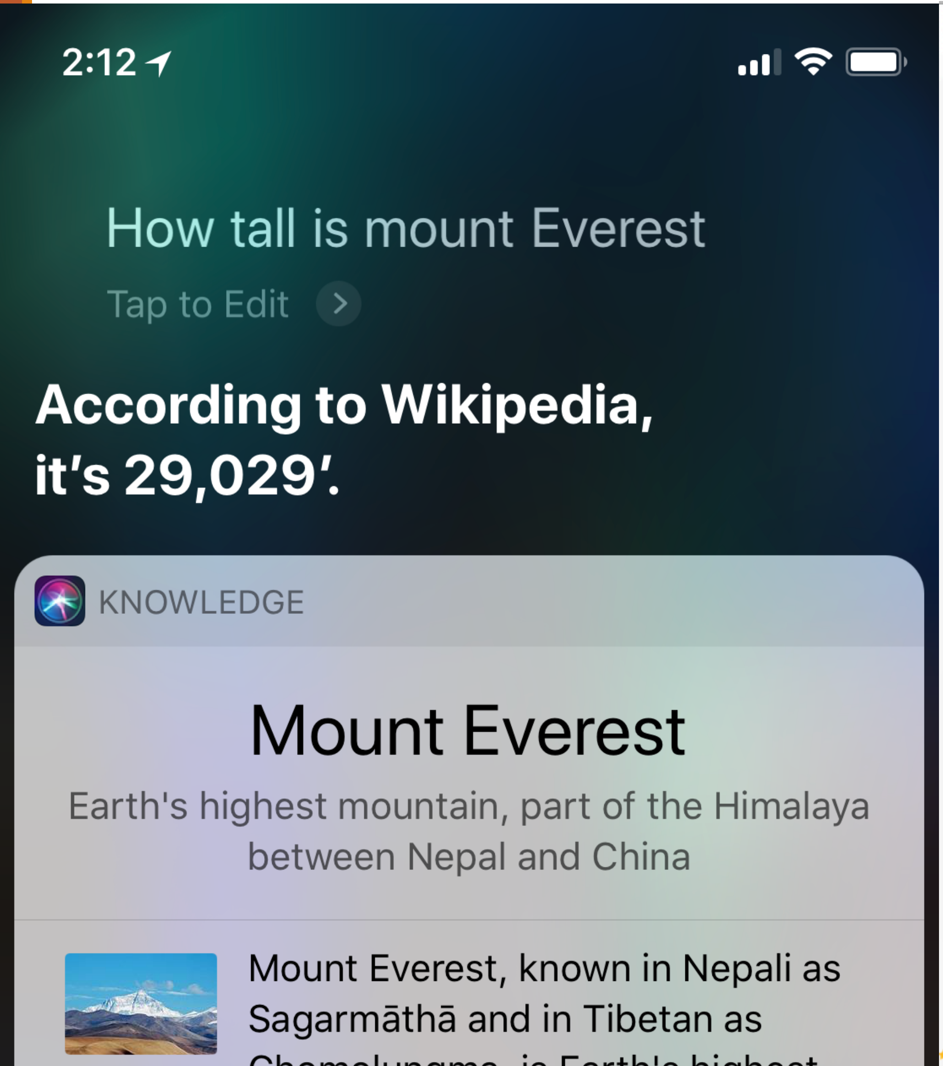
Most words don’t have many **synonyms**, but they do have a lot of **similar** words. *Cat* is not a synonym of *dog*, but *cats* and *dogs* are certainly similar words.

“**fast**” is similar to “**rapid**”

“**tall**” is similar to “**height**”

Useful for applications like question answering

Word similarity



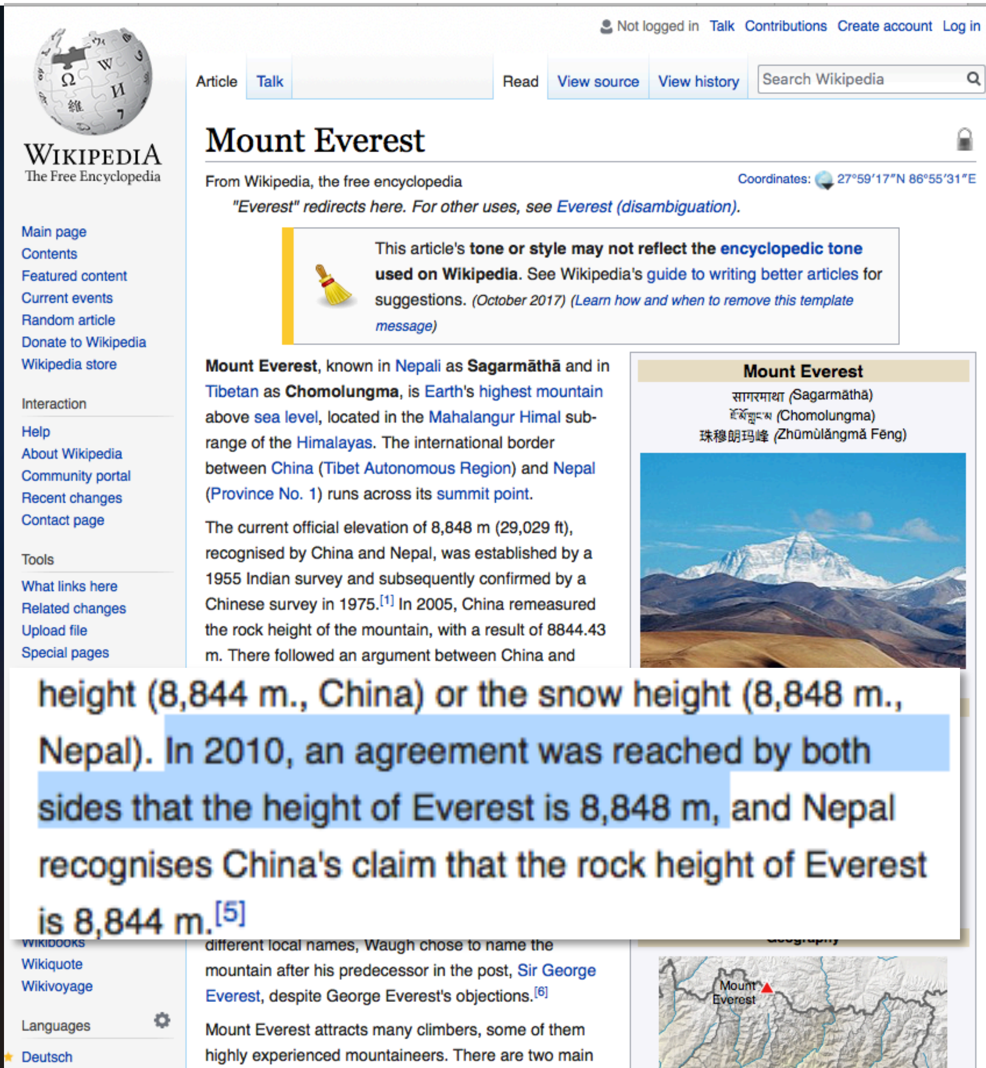
How tall is mount Everest

According to Wikipedia, it's 29,029′.

Mount Everest

Earth's highest mountain, part of the Himalaya between Nepal and China

Mount Everest, known in Nepali as Sagarmāthā and in Tibetan as Chomolungma, is Earth's highest



Mount Everest

From Wikipedia, the free encyclopedia

Coordinates: 27°59′17″N 86°55′31″E

"Everest" redirects here. For other uses, see Everest (disambiguation).

This article's **tone or style may not reflect the encyclopedic tone used on Wikipedia**. See Wikipedia's guide to writing better articles for suggestions. *(October 2017) (Learn how and when to remove this template message)*

Mount Everest, known in Nepali as **Sagarmāthā** and in Tibetan as **Chomolungma**, is Earth's highest mountain above sea level, located in the Mahalangur Himal sub-range of the Himalayas. The international border between China (Tibet Autonomous Region) and Nepal (Province No. 1) runs across its summit point.

The current official elevation of 8,848 m (29,029 ft), recognised by China and Nepal, was established by a 1955 Indian survey and subsequently confirmed by a Chinese survey in 1975.[1] In 2005, China remeasured the rock height of the mountain, with a result of 8844.43 m. There followed an argument between China and

height (8,844 m., China) or the snow height (8,848 m., Nepal). In 2010, an agreement was reached by both sides that the height of Everest is 8,848 m, and Nepal recognises China's claim that the rock height of Everest is 8,844 m.[5]

different local names, Waugh chose to name the mountain after his predecessor in the post, Sir George Everest, despite George Everest's objections.[6]

Mount Everest attracts many climbers, some of them highly experienced mountaineers. There are two main

Mount Everest

सगरमाथा (Sagarmāthā)
ཇོ་མོ་གླང་མ (Chomolungma)
珠穆朗玛峰 (Zhūmùlǎngmǎ Fēng)

Word similarity

Can similar words be substituted in any sentence without changing its truth conditions? No.

How can we measure whether words are similar? One way is to ask humans to judge how similar one word is to another.

Word 1	Word 2	Similarity Score
Vanish	Disappear	9.8
Tiger	Cat	7.4
Love	Sex	6.8
Muscle	Bone	3.6
Cucumber	Professor	0.3

Word Relatedness

Words can still be **related** in ways other than being similar to each other.

Coffee and *Cup* are **not similar** because they don’t share any features

1. *coffee* is a plant or a beverage,
2. *cup* is a manufactured object made in a useful shape

But they’re **related** by co-participating in the same **event.**

Relatedness is measured with **word association** tests in psychology.

A **semantic field** is a set of words which cover a semantic domain and bear structured relations with each other.

Hospitals: *surgeon, scalpel, nurse, anesthetic, hospital*
Restaurants: *waiter, menu, plate, food, chef*
Houses: *family, door, roof, kitchen, bed*

Semantic Roles

An **event** like a commercial transaction described with different **verbs**

1. *buy* (the event from the perspective of the buyer),
2. *sell* (from the perspective of the seller),
3. *pay* (focusing on the monetary aspect),

Or with nouns like *buyer*.

Frames encode **semantic roles** (like *buyer, seller, goods, money*), and the words in a sentence that take on these roles.

Connotation

Words have **affective meanings** or **connotations.** Three important dimensions of affective meaning.

1. *Valence* – the pleasantness of the stimulus
2. *Arousal* – the intensity of emotion provoked by the stimulus
3. *Dominance* – the degree of control exerted by the stimulus

	Valence	Arousal	Dominance
courageous	8.05	5.5	7.38
music	7.67	5.57	6.5
heartbreak	2.45	5.65	3.58
cub	6.71	3.95	4.24
life	6.68	5.59	5.89

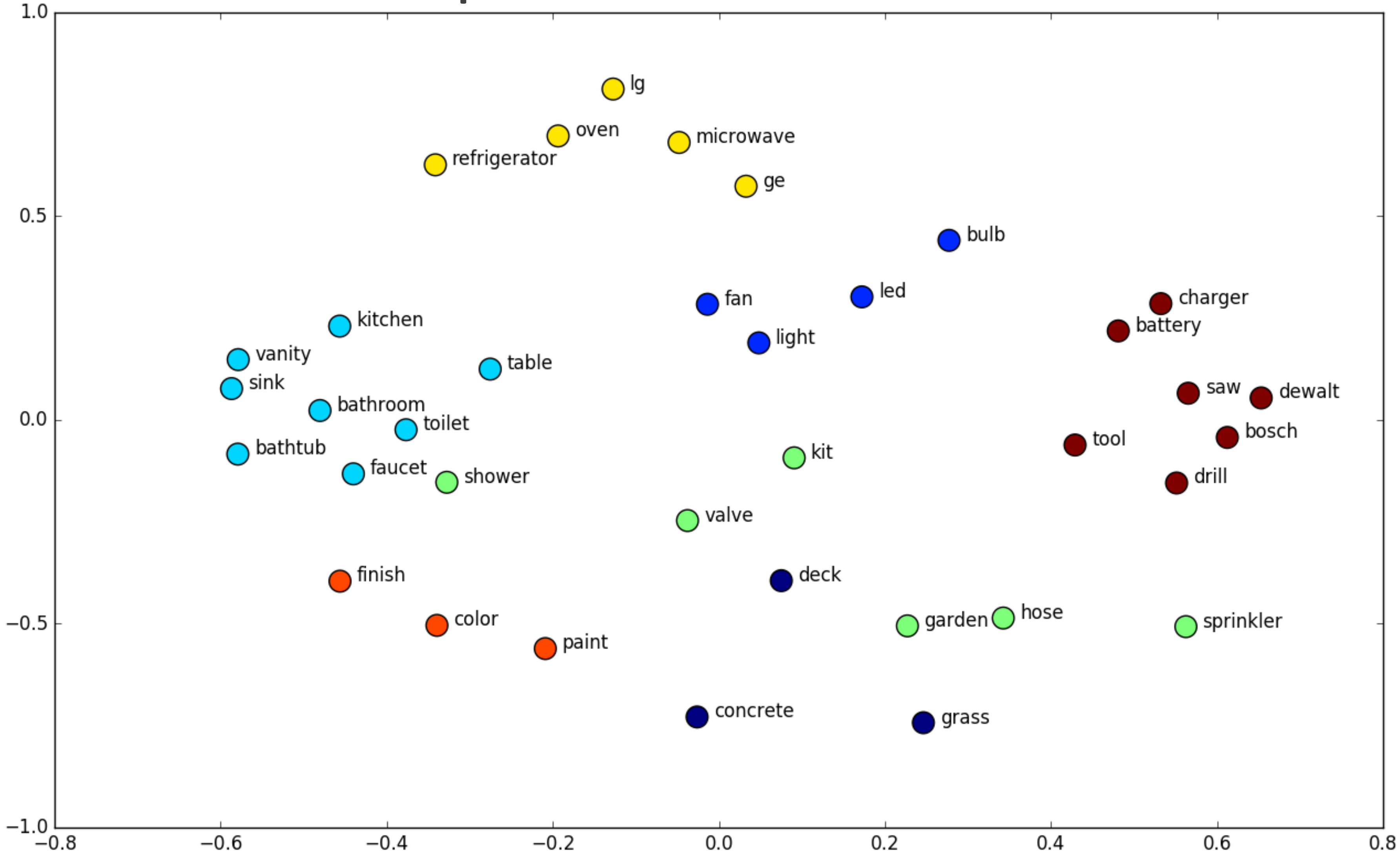
Points in space

Osgood et al. (1957) noticed that in using these 3 numbers to represent the meaning of a word, the model was representing each word as a point in a three-dimensional space

Part of the meaning of *heartbreak* can be represented as a vector with three dimensions corresponded to the word's rating on the three scales.

heartbreak	2.45	5.65	3.58

Vector Space Models



Distributional Hypothesis

If we consider *optometrist* and *eye-doctor* we find that, as our corpus of utterances grows, these two occur in almost the same environments. In contrast, there are many sentence environments in which *optometrist* occurs but *lawyer* does not...

It is a question of the relative frequency of such environments, and of what we will obtain if we ask an informant to substitute any word he wishes for optometrist (not asking what words have the same meaning).

These and similar tests all measure the probability of particular environments occurring with particular elements... If A and B have almost identical environments we say that they are synonyms.
–Zellig Harris (1954)

Intuition of distributional word similarity

Nida (1975) example:

A bottle of **tesgüino** is on the table
Everybody likes **tesgüino**
Tesgüino makes you drunk
We make **tesgüino** out of corn.

From context words humans can guess **tesgüino** means
an alcoholic beverage like beer

Intuition for algorithm:
Two words are similar if they have similar word contexts.

Information Retrieval

- Vector Space Models were initially developed in the SMART information retrieval system (Salton, 1971)
- Each document in a collection is represented as point in a space (a vector in a vector space)
- A user's query is a pseudo-document and is represented as a point in the same space as the documents
- Perform IR by retrieving documents whose vectors are close together in this space to the query vector

Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

Each column vector represents a Document

Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

Each row vector
represents a Term

Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

The value in a cell is based on how often that term occurred in that document

Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

The length of the document vectors is the size of the vocabulary

Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

Document vectors
can be sparse
(most values are 0)

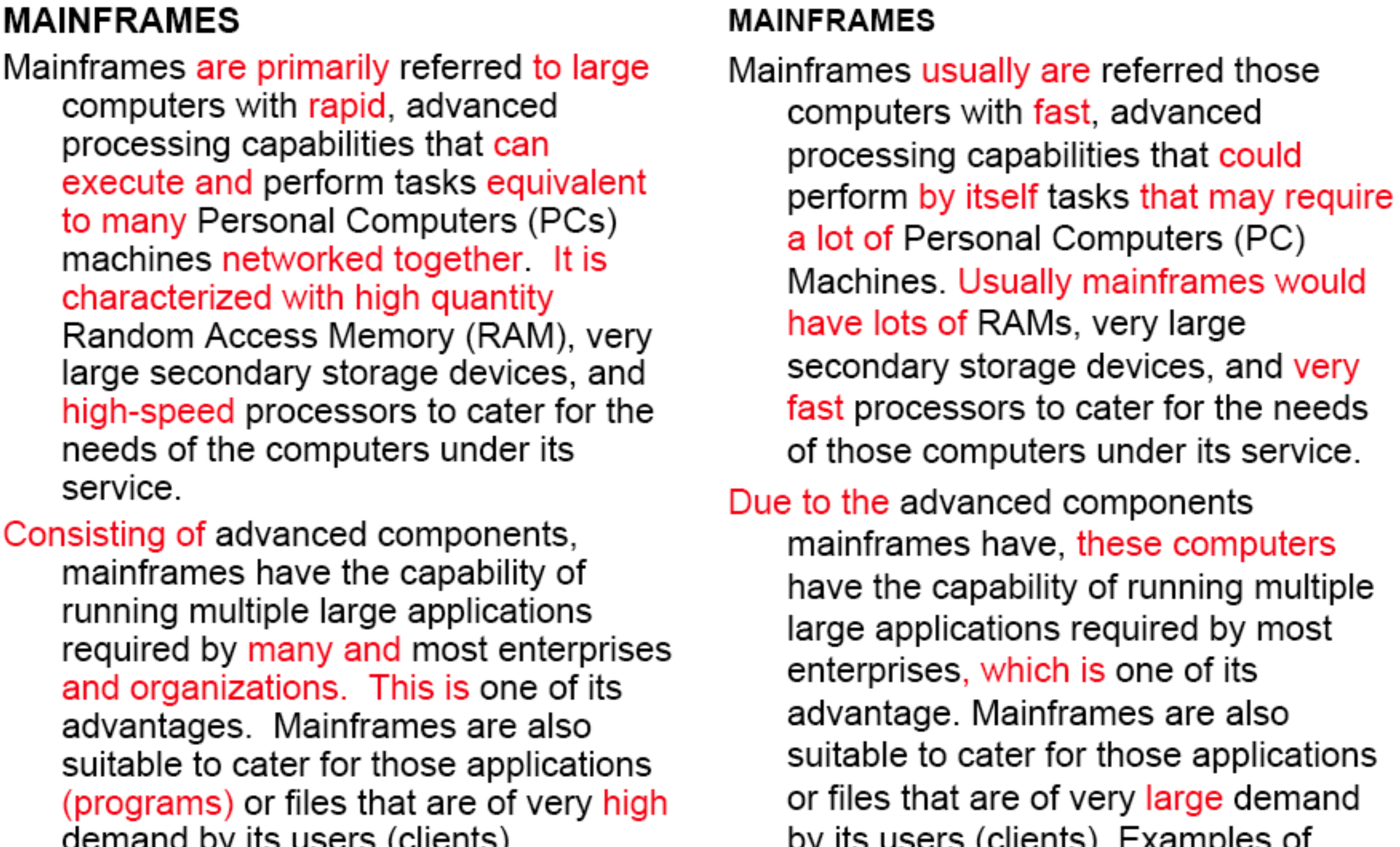
Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

We can measure
how similar two
documents are
by comparing their
column vectors

What can document similarity let you do?

Word similarity for plagiarism detection



MAINFRAMES

Mainframes are primarily referred to large computers with rapid, advanced processing capabilities that can execute and perform tasks equivalent to many Personal Computers (PCs) machines networked together. It is characterized with high quantity Random Access Memory (RAM), very large secondary storage devices, and high-speed processors to cater for the needs of the computers under its service.

Consisting of advanced components, mainframes have the capability of running multiple large applications required by many and most enterprises and organizations. This is one of its advantages. Mainframes are also suitable to cater for those applications (programs) or files that are of very high demand by its users (clients)

MAINFRAMES

Mainframes usually are referred those computers with fast, advanced processing capabilities that could perform by itself tasks that may require a lot of Personal Computers (PC) Machines. Usually mainframes would have lots of RAMs, very large secondary storage devices, and very fast processors to cater for the needs of those computers under its service.

Due to the advanced components mainframes have, these computers have the capability of running multiple large applications required by most enterprises, which is one of its advantage. Mainframes are also suitable to cater for those applications or files that are of very large demand by its users (clients). Examples of

Term-Document Matrix

	D1	D2	D3	D4	D5
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

What does comparing two row vectors do?

Vector comparisons

	docX	docY
A	2	4
B	10	15
C	14	10

Vector comparisons

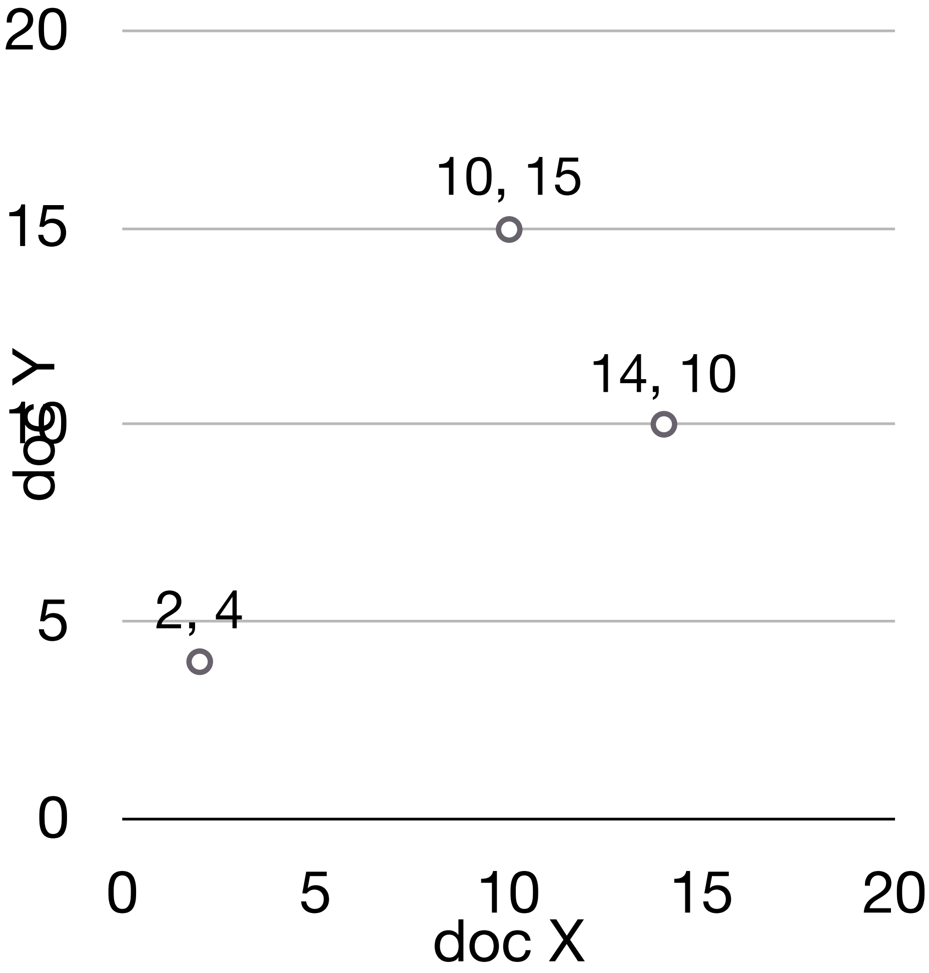
	docX	docY
A	2	4
B	10	15
C	14	10

docY is a positive movie review
docX is a less positive movie review

A = "superb"	positive / low frequency
B = "good"	positive / high frequency
C = "disappointing"	negative / high frequency

Vector comparisons

	docX	docY
A	2	4
B	10	15
C	14	10

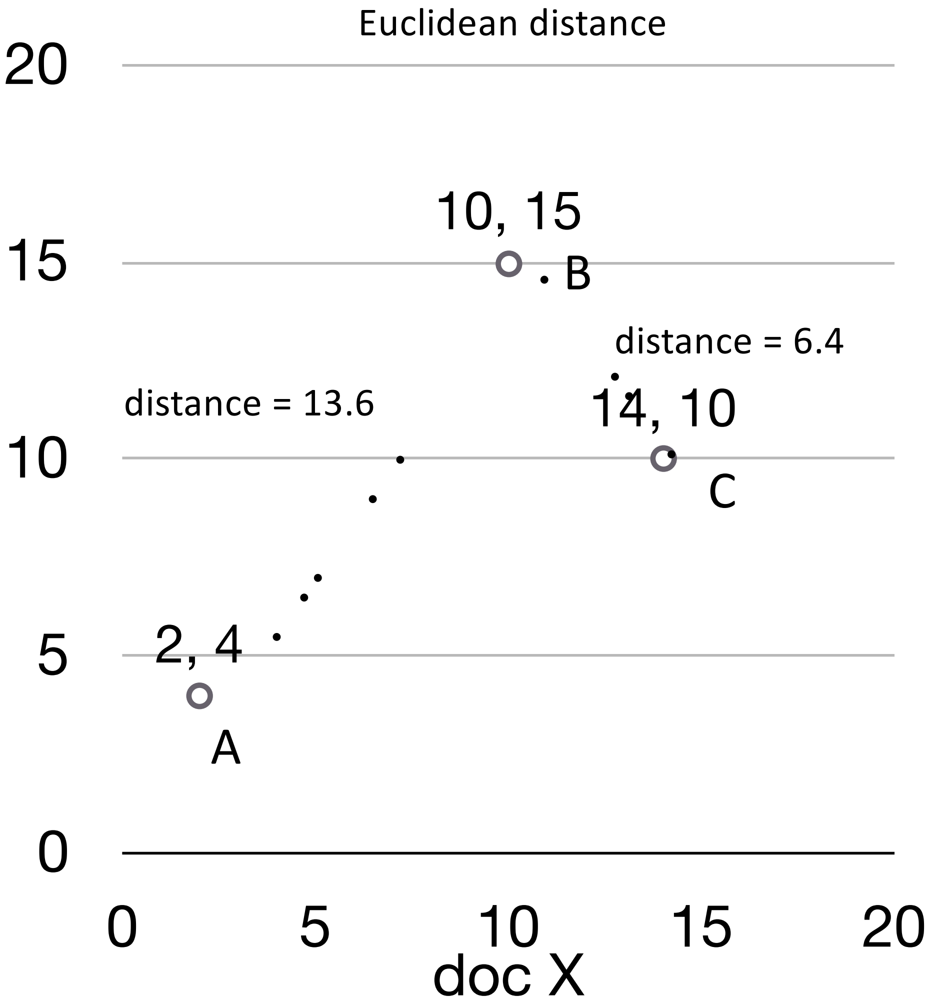


Vector comparisons

	docX	docY
A	2	4
B	10	15
C	14	10

Euclidean distance : vectors u, v of dimension N

$$\sqrt{\sum_{i=1}^{N} |u_i - v_i|^2}$$



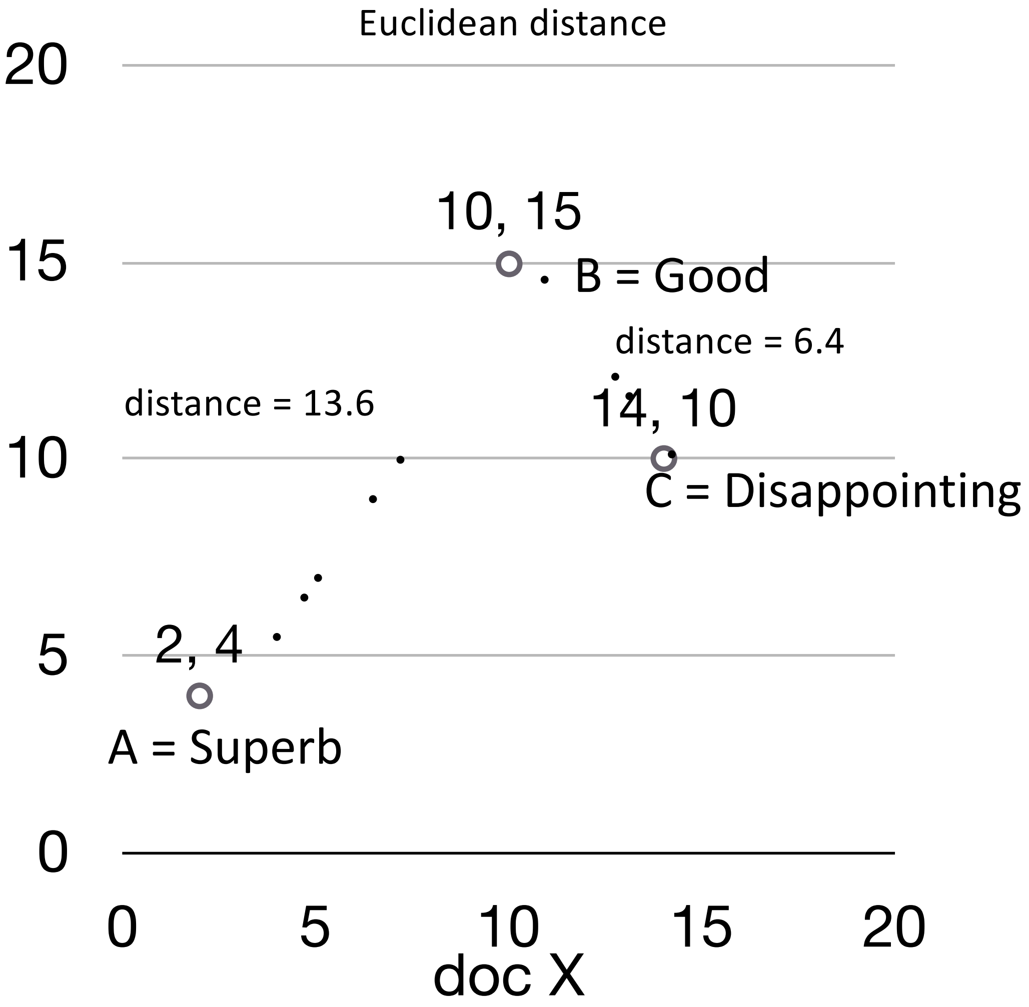
Vector comparisons

Oh no! Good is closer to Disappointing than to Superb.

	docX	docY
A	2	4
B	10	15
C	14	10

Euclidean distance : vectors u, v of dimension N

$$\sqrt{\sum_{i=1}^{N} |u_i - v_i|^2}$$



Vector L2 (length) Normalization

	docX	docY
A	2	4
B	10	15
C	14	10

$\lVert u \rVert$
4.47
18.02
17.20

$$\lVert u \rVert = \sqrt{\textstyle\sum_{i=1}^{n} u_i^2}$$

Vector L2 (length) Normalization

	docX	docY
A	2/4.47	4/4.47
B	10/18.02	15/18.02
C	14/17.2	10/17.2

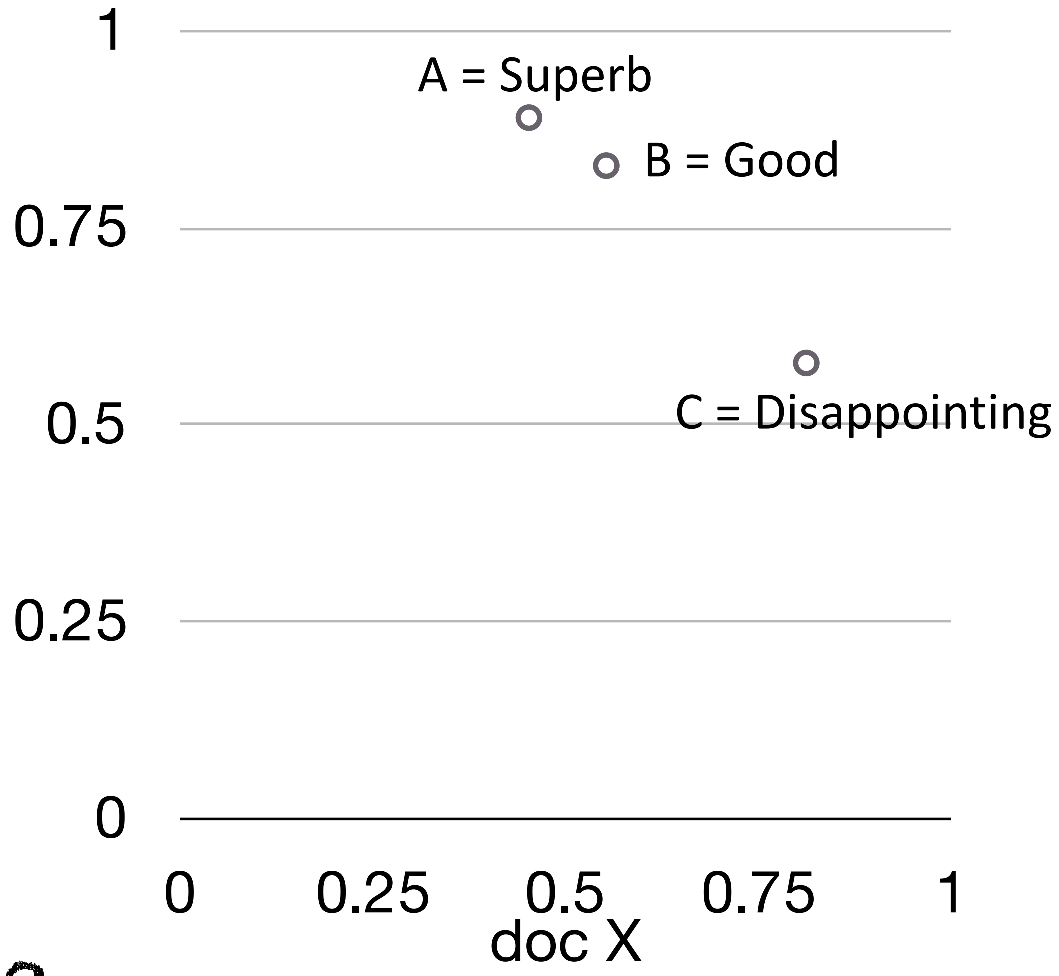
||u||
4.47
18.02
17.20

$$\|u\| = \sqrt{\Sigma_{i=1}^{n} u_i^2}$$

Divide each vector by its L2 length

Vector L2 (length) Normalization

	docX	docY
Ȧ	0.45	0.89
Ḃ	0.55	0.83
Ċ	0.81	0.58

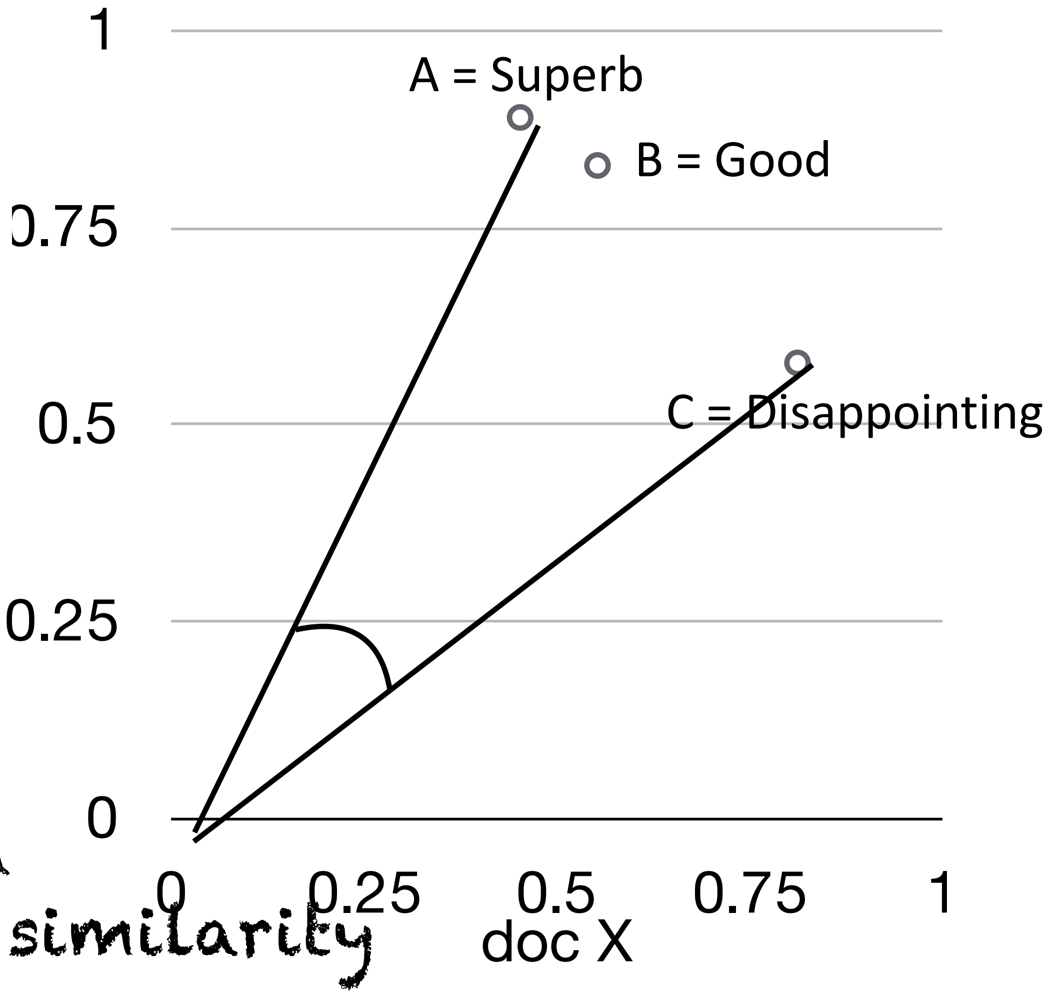


Now Good is closer to Superb than to Disappointing

Cosine Distance

$$1 - \frac{\sum_{i=1}^{n} u_i \times v_i}{\sqrt{\sum_{i=1}^{n} u_i^2} \times \sqrt{\sum_{i=1}^{n} v_i^2}}$$

Cosine does the L2 normalization too



Cosine angle between vectors tells us their similarity

Term-Term Matrix

	abandon	abdicate	abhor	…	zymurgy
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

Term-Term Matrix

AKA
Term-Context
Matrix

	abandon	abdicate	abhor	…	zymurgy
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

Length of the vector is now |V|
instead of number of documents

Term-Term Matrix

AKA
Term-Context
Matrix

back	abandon	abdicate	abhor	…	zymurgy
abandon					
abdicate					
abhor					
academic					
…					
zygodactyl					
zymurgy					

The value in a cell indicates how often abandon appears in a context window surrounding abdicate

Context windows

w-2, w-1 **target_word** w+1 w+2

The government most not **abdicate** responsibility to non-elected

it has led men to **abdicate** their family responsibilities

other demands, but declining to **abdicate** his responsibility

leaders **abdicate** their role and present people with no plans

	his	leaders	not	responsibility	to
abandon	1	1	1	2	3

Context windows

Occur in a window of +/- 2 words, in the same sentence, in the same document

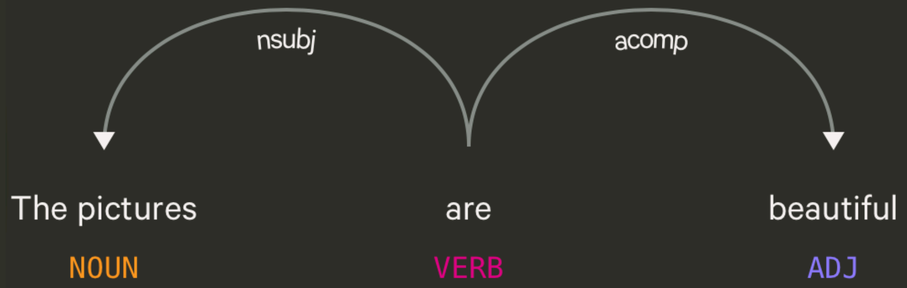
Instead of window of words use more complex contexts: dependency patters. Subj-of-verb, adj-mod, obj-of-verb

Languages have long distance dependencies

The ***pictures are*** *beautiful.*

The ***pictures*** *of the old man* ***are*** *beautiful.*

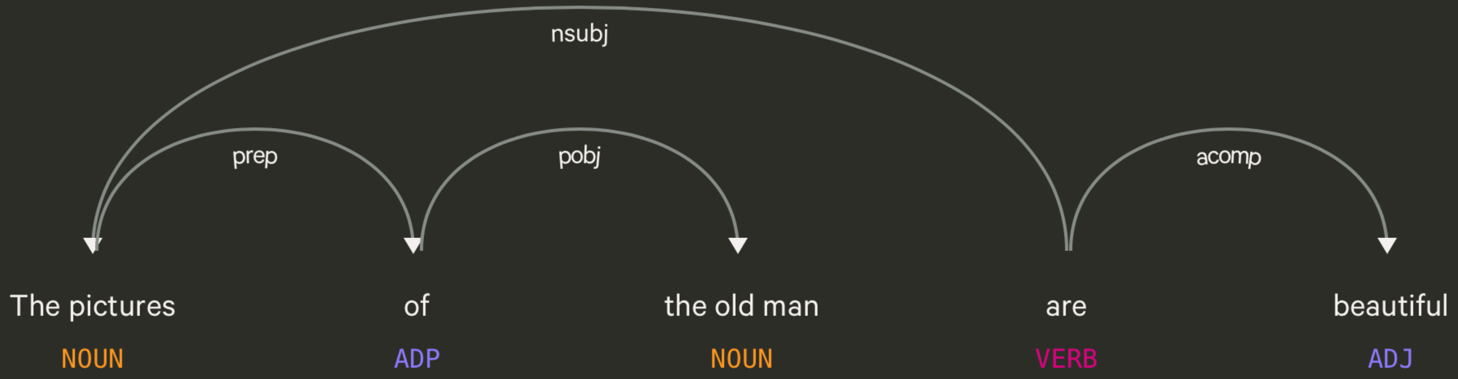
The ***pictures*** *of the old man holding his grandchild* ***are*** *beautiful.*



nsubj acomp

The pictures are beautiful

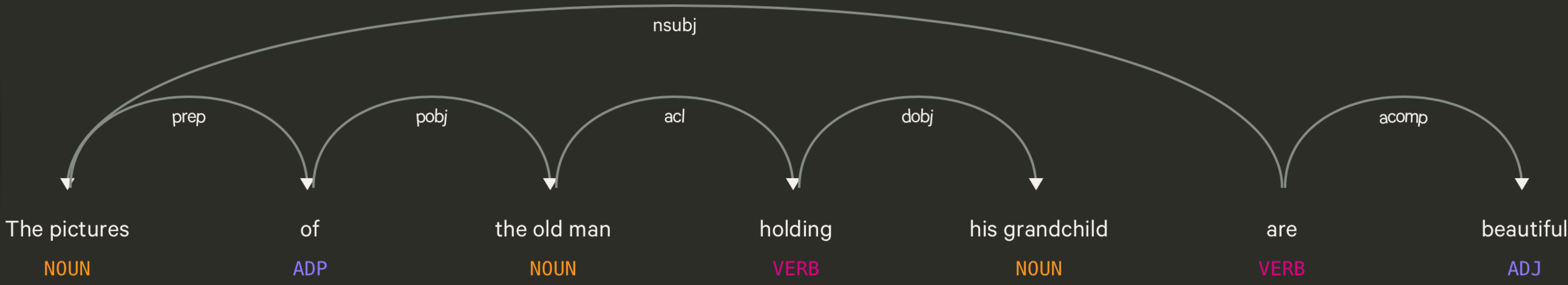
NOUN VERB ADJ



nsubj prep pobj acomp

The pictures of the old man are beautiful

NOUN ADP NOUN VERB ADJ



nsubj prep pobj acl dobj acomp

The pictures of the old man holding his grandchild are beautiful

NOUN ADP NOUN VERB NOUN VERB ADJ

Using syntax to define a word’s context

Zellig Harris (1968) “The meaning of entities, and the meaning of grammatical relations among them, is related to the restriction of combinations of these entities relative to other entities”

Duty and **Responsibility** have similar syntactic distributions

Modified by adjectives	additional, administrative, assumed, collective, congressional, constitutional …
Object of verbs	assert, assign, assume, attend to, avoid, become, breach..

Alternates to counts

Raw word frequency is not a great measure of association between words. It’s very skewed “the” and “of” are very frequent, but maybe not the most discriminative

We’d rather have a measure that asks whether a context word is particularly informative about the target word.

Instead of raw counts, it’s common to transform vectors using TF-IDF or PPMI

TF-IDF

*Term frequency * inverse document frequency*

How often a word occurred in a document

1 over the number of documents that it occurred in

Sparse v. Dense Vectors

Co-occurrence matrix (weighted by TF-IDF or mutual information)

- **Long** (length |V| = 50,000+)
- **Sparse** (most elements are zeros)

Alternative: learn vectors that are

- **Short** (length 200-1000)
- **Dense** (most elements are non-zero)

How do we get dense vectors?

One recipe: train a classifier!

1. Treat the target word and a neighboring context word as positive examples.

2. Randomly sample other words in the lexicon to get negative samples.

3. Use logistic regression (similar to Perceptron, but output values range between 0-1) to train a classifier to distinguish those two cases.

4. Use the **weights** as the **embeddings**.

Skip-grams, CBOW Mikolov et al. 2013

Learn embeddings as part of the process of word prediction.

Train a classifier to predict neighboring words

Inspired by neural net language models.

In so doing, learn dense embeddings for the words in the training corpus.

Advantages:

Fast, easy to train (much faster than SVD)

Available online in the word2vec package

Including sets of pretrained embeddings!

Skip-Grams

Predict each neighboring word in a context window of 2C of surrounding words

So for C=2, we are given a word w_t and we try to predict its 4 surrounding words

$$[w_{t-2}, w_{t-1}, w_{t+1}, w_{t+2}]$$

Uses "negative sampling" for training

Negative sampling

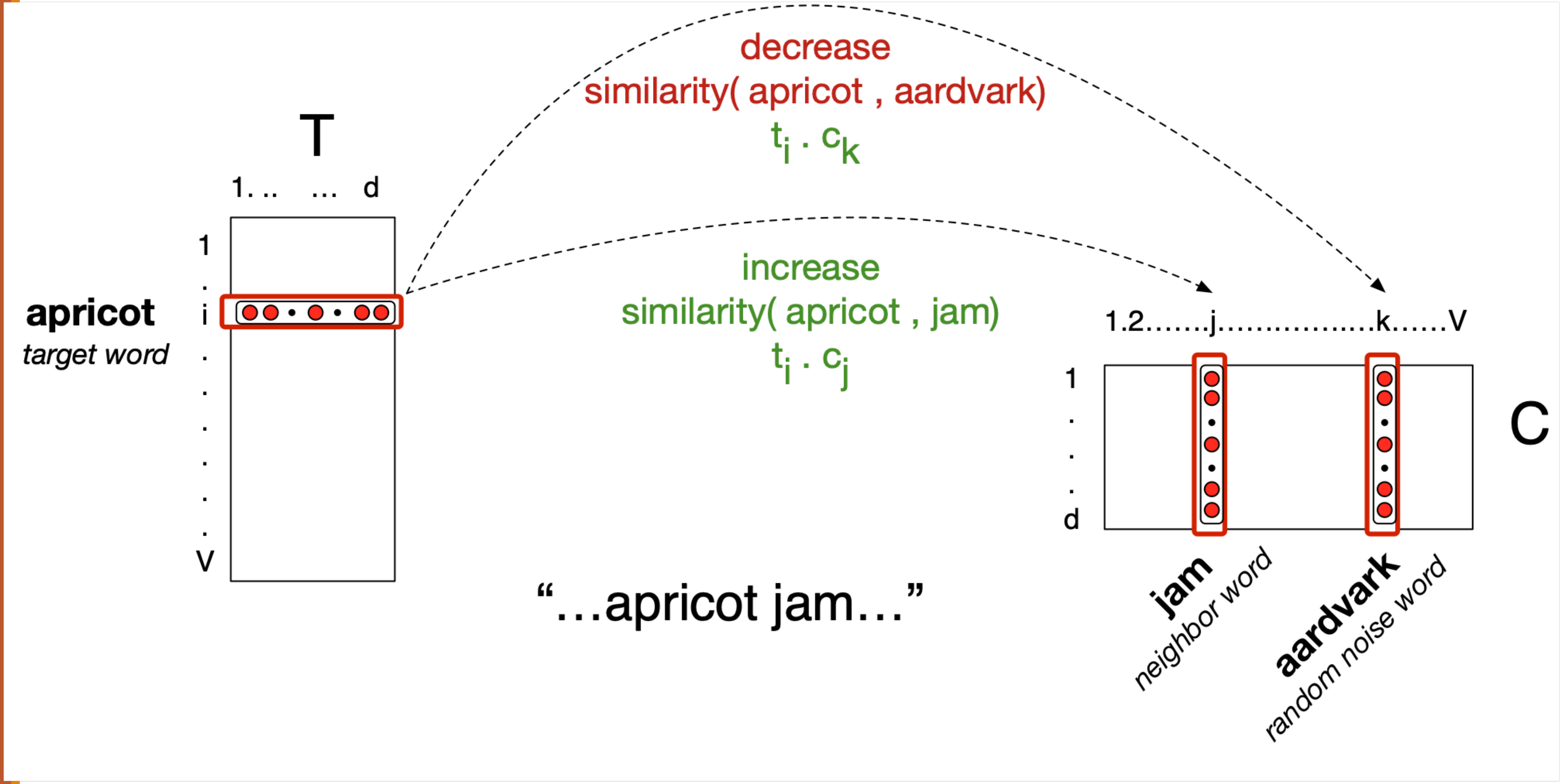
```
lemon,  a [tablespoon of apricot preserves or] jam
           c1         c2    w    c3        c4
```

We want predictions
of these words to be high

And these words to be low

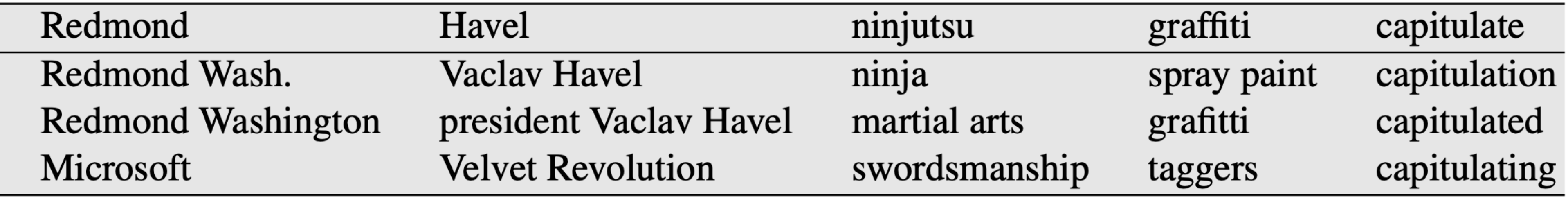
```
[cement metaphysical dear coaxial     apricot attendant whence  forever puddle]
 n1     n2           n3   n4                  n5        n6      n7       n8
```

Neural Network



Properties of Embeddings

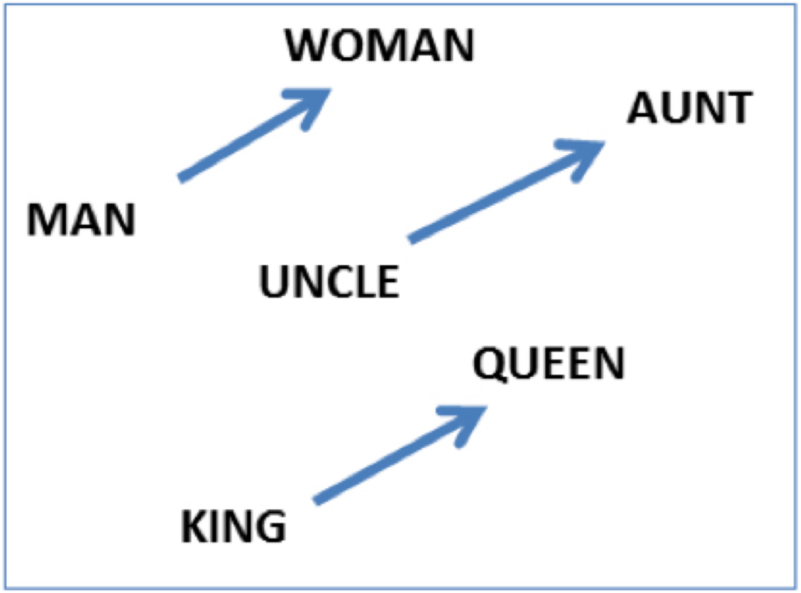
Nearest Neighbors are surprisingly good



Redmond	Havel	ninjutsu	graffiti	capitulate
Redmond Wash.	Vaclav Havel	ninja	spray paint	capitulation
Redmond Washington	president Vaclav Havel	martial arts	grafitti	capitulated
Microsoft	Velvet Revolution	swordsmanship	taggers	capitulating

Embeddings capture relational meanings

.vector('king') - vector('man') + vector('queen') ≅ vector('woman')



Magnitude: A Fast, Efficient Universal Vector Embedding Utility Package

Ajay Patel
Plasticity Inc.
San Francisco, CA
ajay@plasticity.ai

Alexander Sands
Plasticity Inc.
San Francisco, CA
alex@plasticity.ai

Chris Callison-Burch
Computer and Information
Science Department
University of Pennsylvania
ccb@upenn.edu

Marianna Apidianaki
LIMSI, CNRS
Université Paris-Saclay
91403 Orsay, France
marapi@seas.upenn.edu

Abstract

Vector space embedding models like word2vec, GloVe, and fastText are extremely popular representations in natural language processing (NLP) applications. We present Magnitude, a fast, lightweight tool for utilizing and processing embeddings. Magnitude is an open source Python package with a compact vector storage file format that allows for efficient manipulation of huge numbers of embeddings. Magnitude performs common operations up to 60 to 6,000 times faster than Gensim. Magnitude introduces several novel features for improved robustness like

Metric	Cold	Warm
Initial load time	97x	–
Single key query	1x	110x
Multiple key query (n=25)	68x	3x
k-NN search query (k=10)	1x	5,935x

Table 1: Speed comparison of Magnitude versus Gensim for common operations. The 'cold' column represents the first time the operation is called. The 'warm' column indicates a subsequent call with the same keys.

file, a 97x speed-up. Gensim uses 5GB of RAM versus 18KB for Magnitude.

Demo of word vectors

```
# Install Magnitude
pip3 install pymagnitude

# Download Google’s word2vec vectors
wget http://magnitude.plasticity.ai/word2
# Warning it’s 11GB large

# Start Python, and try the commands
# on the next slide
python3
```

Demo of word vectors

```
from pymagnitude import *
vectors = Magnitude("GoogleNews-vectors-ne

queen = vectors.query('queen')
king = vectors.query("king")
vectors.similarity(king, queen)
# 0.6510958

vectors.most_similar_approx(king, topn=5)
#[('king', 1.0), ('kings', 0.72), ('prince
```

Many possible models

Matrix type
Term-document
Term-context
Pattern-pair

Dim. Reduction
word2vec
GloVe
PCA
LDA
LSA

Reweighting
length norm.
TF-IDF
PPMI
probabilities

Comparisons
cosine
Manhattan
Jaccard
KL divergence
JS distance
DICE

How many dimensions?

What modifications should we make to the input?