
Ceci n'est pas une pipe.
Magritte

WORD
VECTOR SPACE MODELS

The Treachery of Images

Artist	René Magritte
Year	1929
Medium	Oil on canvas
Movement	Surrealism
Dimensions	60.33 cm × 81.12 cm (23.75 in × 31.94 in)
Location	Los Angeles County Museum of Art[1]

Cosine Similarity

the angle ∈ [0,360], Cosine Similarity ∈ [-1, 1]

$$\text{cosine}(\mathbf{v},\mathbf{w}) = \frac{\mathbf{v}\cdot\mathbf{w}}{|\mathbf{v}||\mathbf{w}|} = \frac{\sum_{i=1}^{N} v_i w_i}{\sqrt{\sum_{i=1}^{N} v_i^2}\sqrt{\sum_{i=1}^{N} w_i^2}}$$

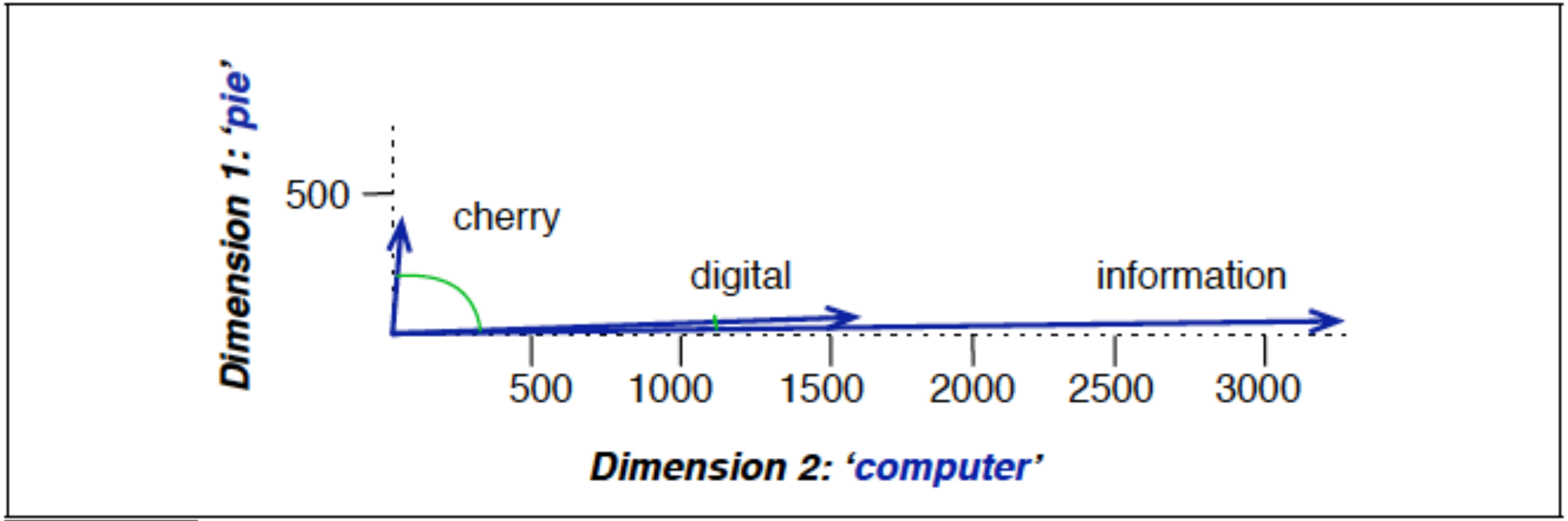
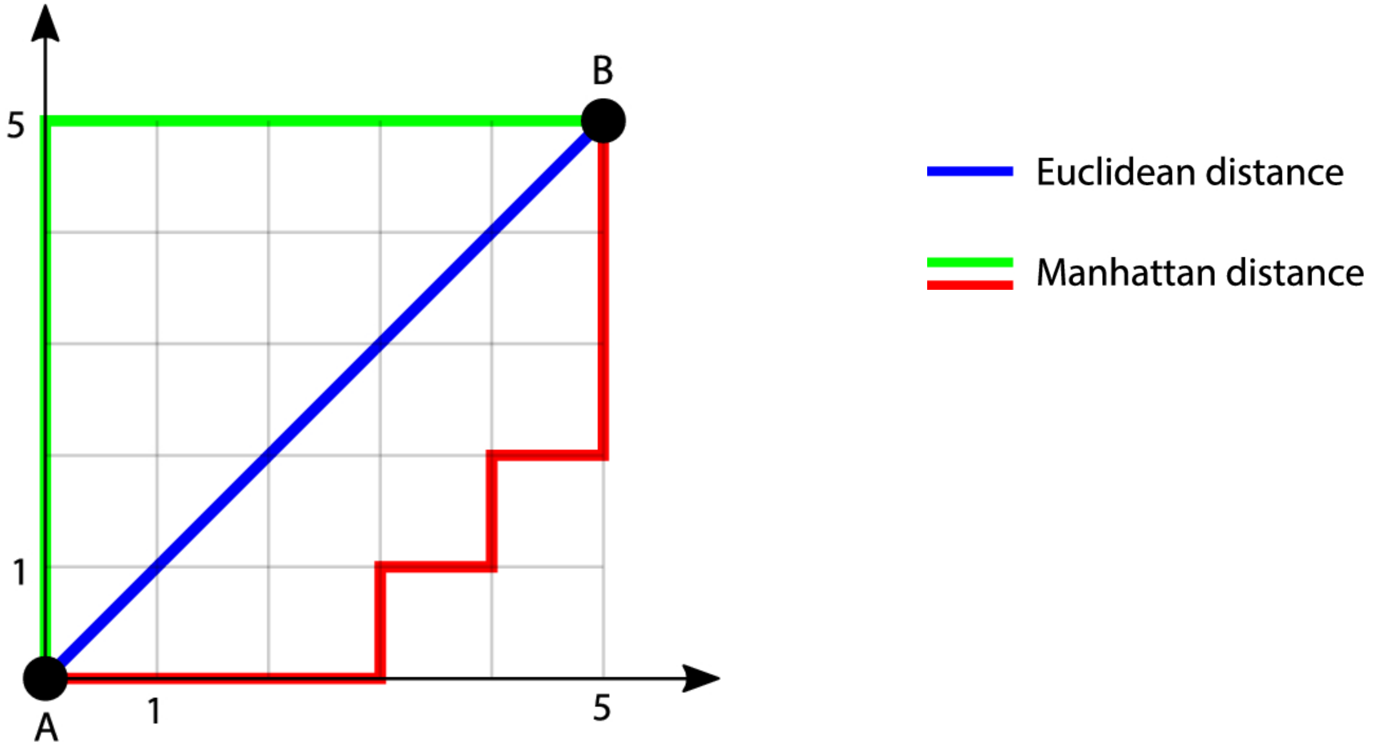


Figure 6.7 A (rough) graphical demonstration of cosine similarity, showing vectors for three words (*cherry*, *digital*, and *information*) in the two dimensional space defined by counts of the words *computer* and *pie* nearby. Note that the angle between *digital* and *information* is smaller than the angle between *cherry* and *information*. When two vectors are more similar, the cosine is larger but the angle is smaller; the cosine has its maximum (1) when the angle between two vectors is smallest (0°); the cosine of all other angles is less than 1.

Minkowski Distance

$$\left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{1/p}$$

p = 1, Manhattan Distance
p = 2, Euclidean Distance
p = ∞, Chebychev Distance



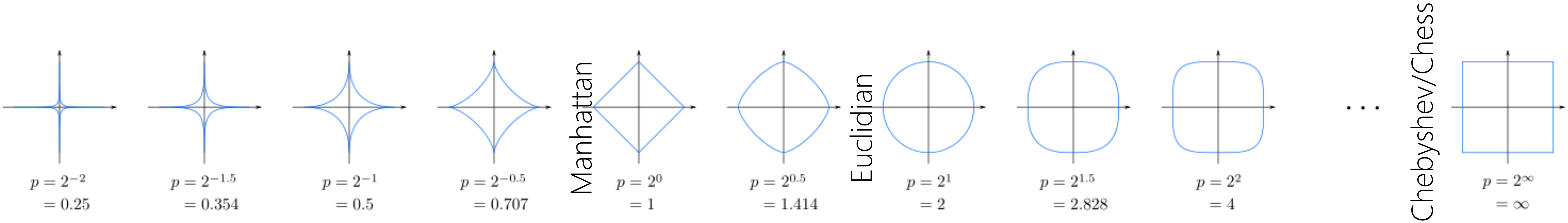
$$\left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{1/p}$$

p = 1, Manhattan Distance

p = 2, Euclidean Distance

Minkowski Distance

Blue lines show all points (x,y) with same distance to the center (0,0)



Does it matter? Why? How if yes?

Research Question (RQ)

Cosine Similarity

Minkowski Distance

Vector Semantics

Sparse vs. Dense

Method	Size of word/token/term vector	Sparse/Dense
Word-Documents (TF)	|Documents|	Sparse (Integer)
Term-Term	|Vocabs|	Sparse (Integer)
TF-iDF	|Vocabs|	Sparse (Real)
PMI	|Vocabs|	Sparse (Real)
?	10, 100, …	Dense (Real)

Vector Semantics
Sparse vs. Dense

Dense vectors work better in every NLP task than sparse vectors. Why? We don't completely understand!

Some guesses:

- Dense vectors lead to a model with less parameter: 100-D vs. 50,000-D vectors for a simple binary classifier
  - Generalize better
  - Avoid overfitting
- Captures word semantic dependencies
  - Do a better job of capturing synonymy than sparse vectors.
  - In word space, each dimension is a word. However, these dimensions may not be independent!

Dimensionality Reduction

Drop less informative dimensions (columns)

- Stop-words

Matrix Factorization (Decomposition)

- SVD (Eigenvalues, change of base to eigenvectors, …)

Predictive Models
Word2Vec

Predictive Models

Given a context: … [tablespoon of apricot jam, a] …

- Choose a word as target word *t:* apricot
- Choose others as context word $c_{i:}$ jam, tablespoon

Estimate d-dimensional vectors for *t* and all c_i

- Such that they are close to each other in d-dimensional space
- where d ≪ |Vocabs| or |Documents|

Word2Vec → close → $\sigma(V_t \cdot V_{c_i}) = \dfrac{1}{1+e^{-(V_w \cdot V_{c_i})}}$

Predictive Models

Given a context: … [tablespoon of apricot jam, a] …

- Choose a word as target word *t:* apricot
- Choose random word n_i from out of context: car, phone, …

Estimate d-dimensional vectors for *t* and all n_i

- Such that they are far from each other in d-dimensional space
- where $d \ll$ |Vocabs| or |Documents|

Word2Vec → distant → $\sigma(V_t \cdot -V_{n_i}) = \frac{1}{1+e^{+(V_w \cdot V_{n_i})}}$

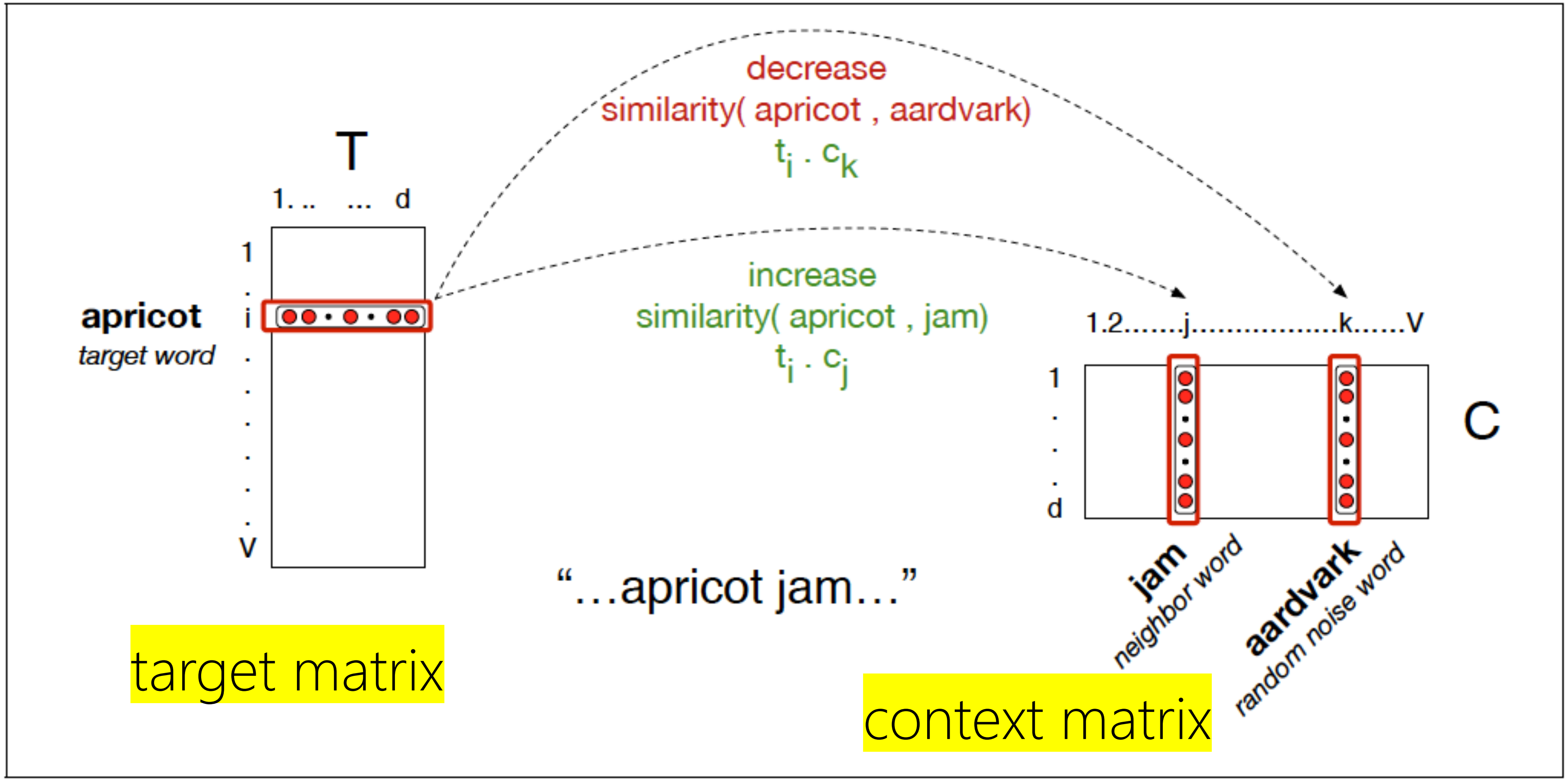


Figure 6.12 The skip-gram model tries to shift embeddings so the target embeddings (here for *apricot*) are closer to (have a higher dot product with) context embeddings for nearby words (here *jam*) and further from (have a lower dot product with) context embeddings for words that don't occur nearby (here *aardvark*).

Is it possible to use only one matrix?

Word2Vec

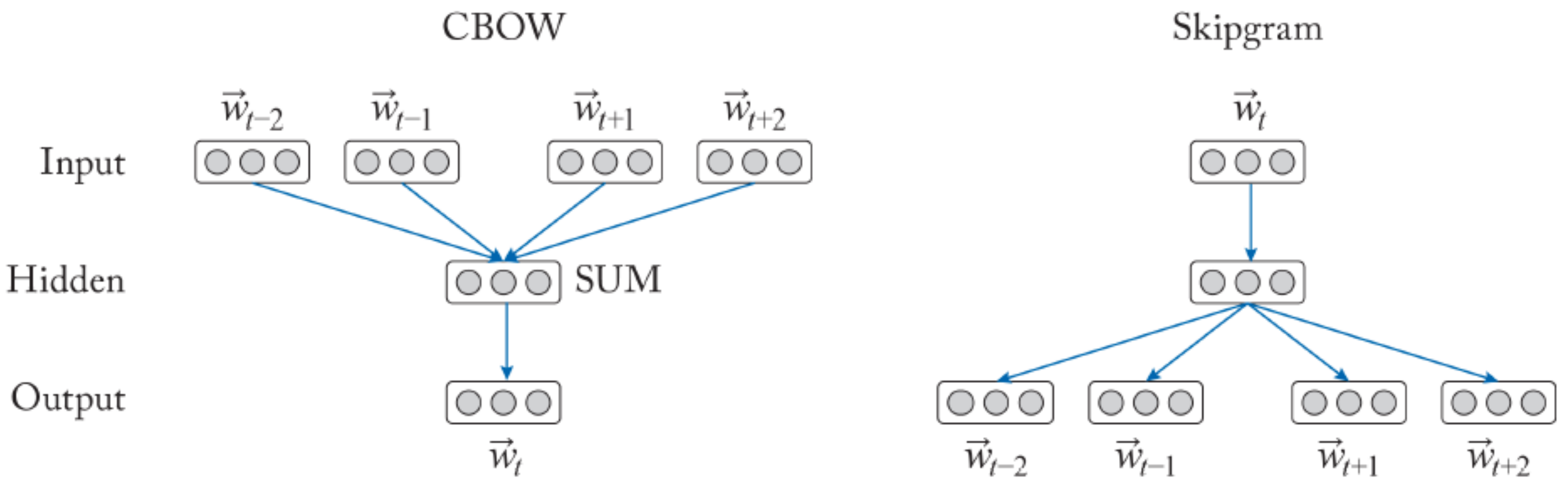
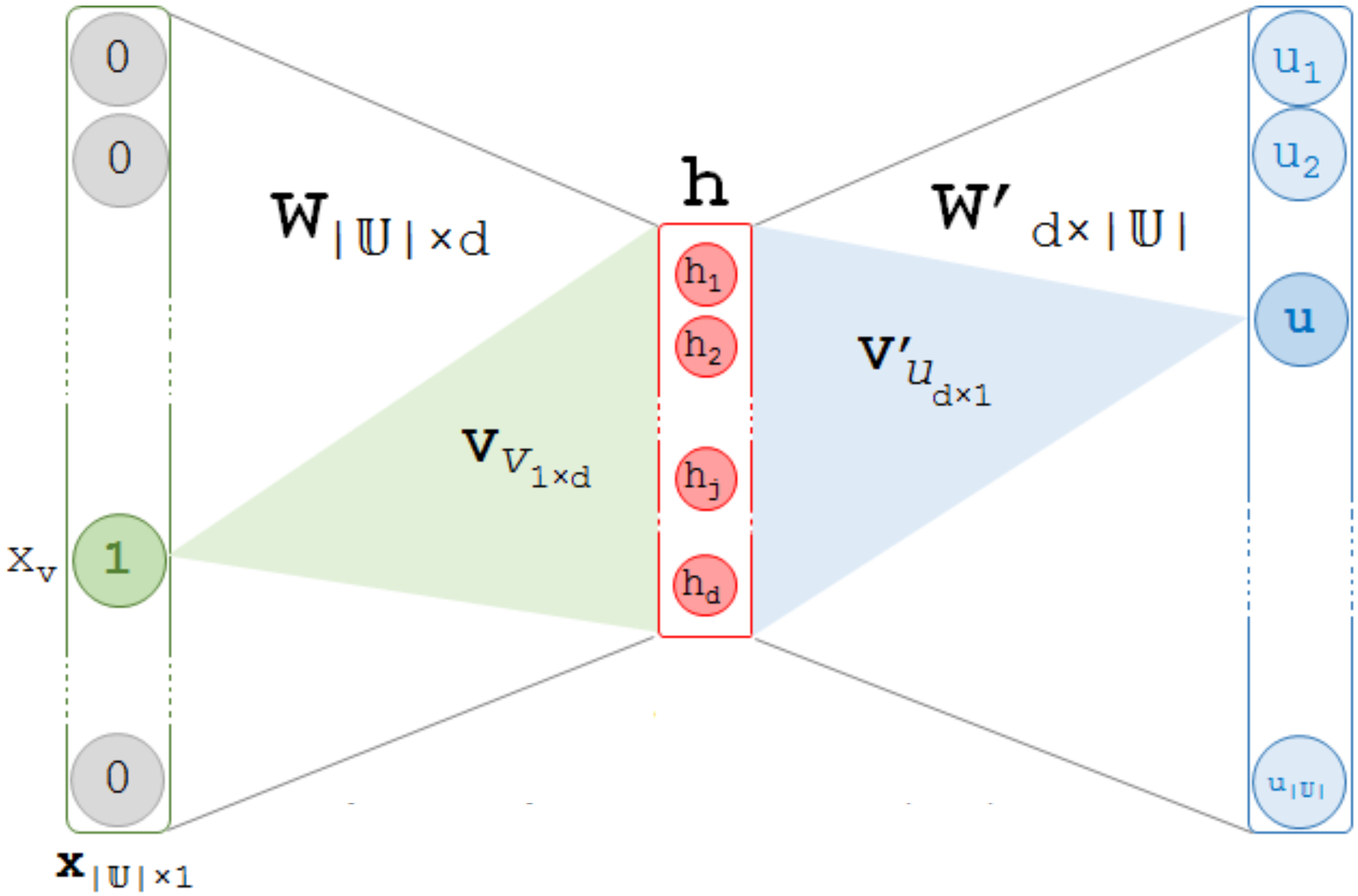


Figure 3.1: Learning architecture of the CBOW and Skip-gram models of Word2vec [Mikolov et al., 2013a].

Word2Vec

$$\sigma\left(\left(\mathbf{h} = \mathbf{x}^{T}\,\mathbf{W} + \cancel{\mathbf{b}}\right)\mathbf{W}' + \cancel{\mathbf{b}}\right)$$



Word2Vec

$$P(+|t,c) = \frac{1}{1+e^{-t\cdot c}}$$

$$\begin{aligned} P(-|t,c) &= 1 - P(+|t,c) \\ &= \frac{e^{-t\cdot c}}{1+e^{-t\cdot c}} \end{aligned}$$

Independent Assumption: P(x,y) = p(x)p(y)

$$\begin{aligned} P(+|t,c_{1:k}) &= \prod_{i=1}^{k} \frac{1}{1+e^{-t\cdot c_i}} \\ \log P(+|t,c_{1:k}) &= \sum_{i=1}^{k} \log \frac{1}{1+e^{-t\cdot c_i}} \end{aligned}$$

$$L(\theta) = \sum_{(t,c)\in +} \log P(+|t,c) + \sum_{(t,c)\in -} \log P(-|t,c)$$

$$\begin{aligned} L(\theta) &= \log P(+|t,c) + \sum_{i=1}^{k} \log P(-|t,n_i) \\ &= \log \sigma(c\cdot t) + \sum_{i=1}^{k} \log \sigma(-n_i \cdot t) \\ &= \log \frac{1}{1+e^{-c\cdot t}} + \sum_{i=1}^{k} \log \frac{1}{1+e^{n_i\cdot t}} \end{aligned}$$

Word2Vec

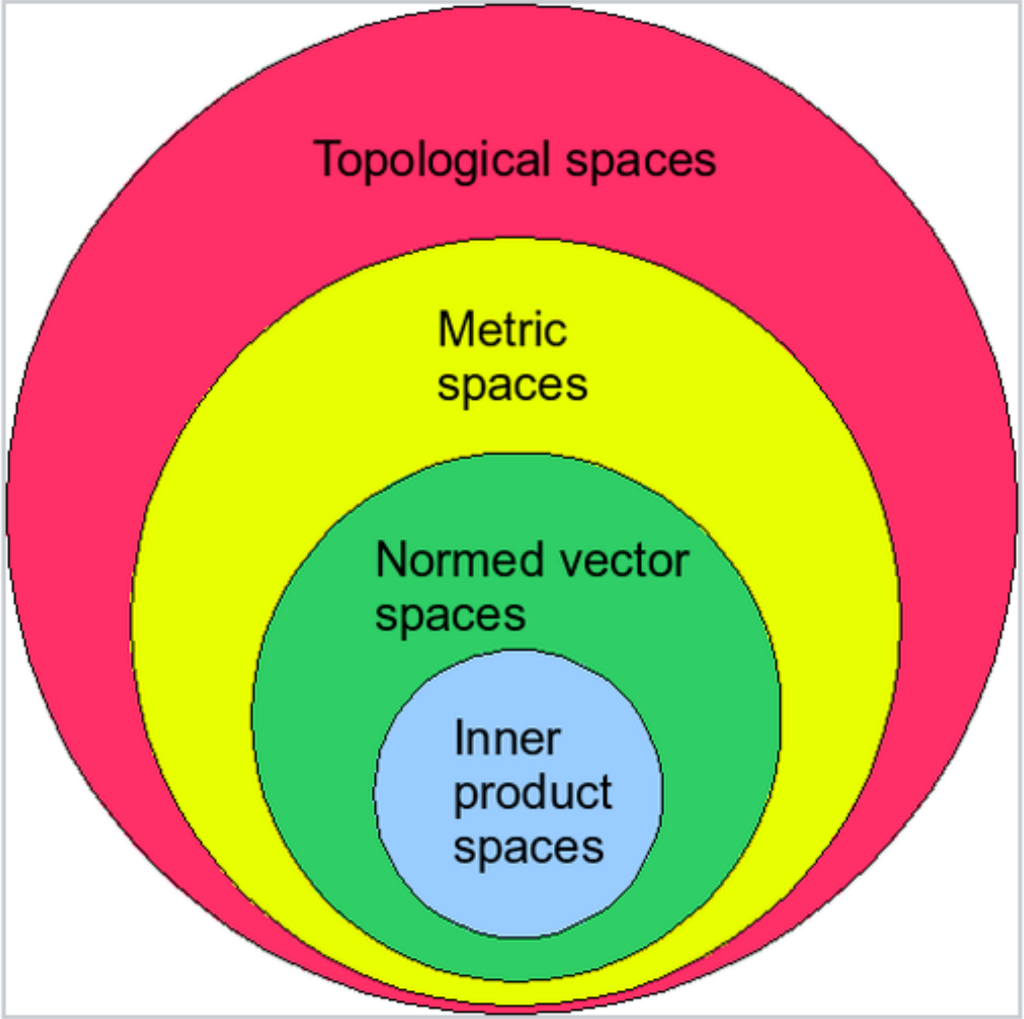
- **Context Window?** Longer vs. Shorter?
- **Deterministic?** Any runs of training ended with same set of vectors?
- **Transformation?** rotation, flips, shear (skew), ...
- **Which signifier:**
  1. [cat], [miu], [*image_of_cat*], [ascii_cat],
  2. Count-based: [*tf*], [*tf-idf*], ...
  3. Learning methods: [word2vec]

Pre-trained Word Vectors

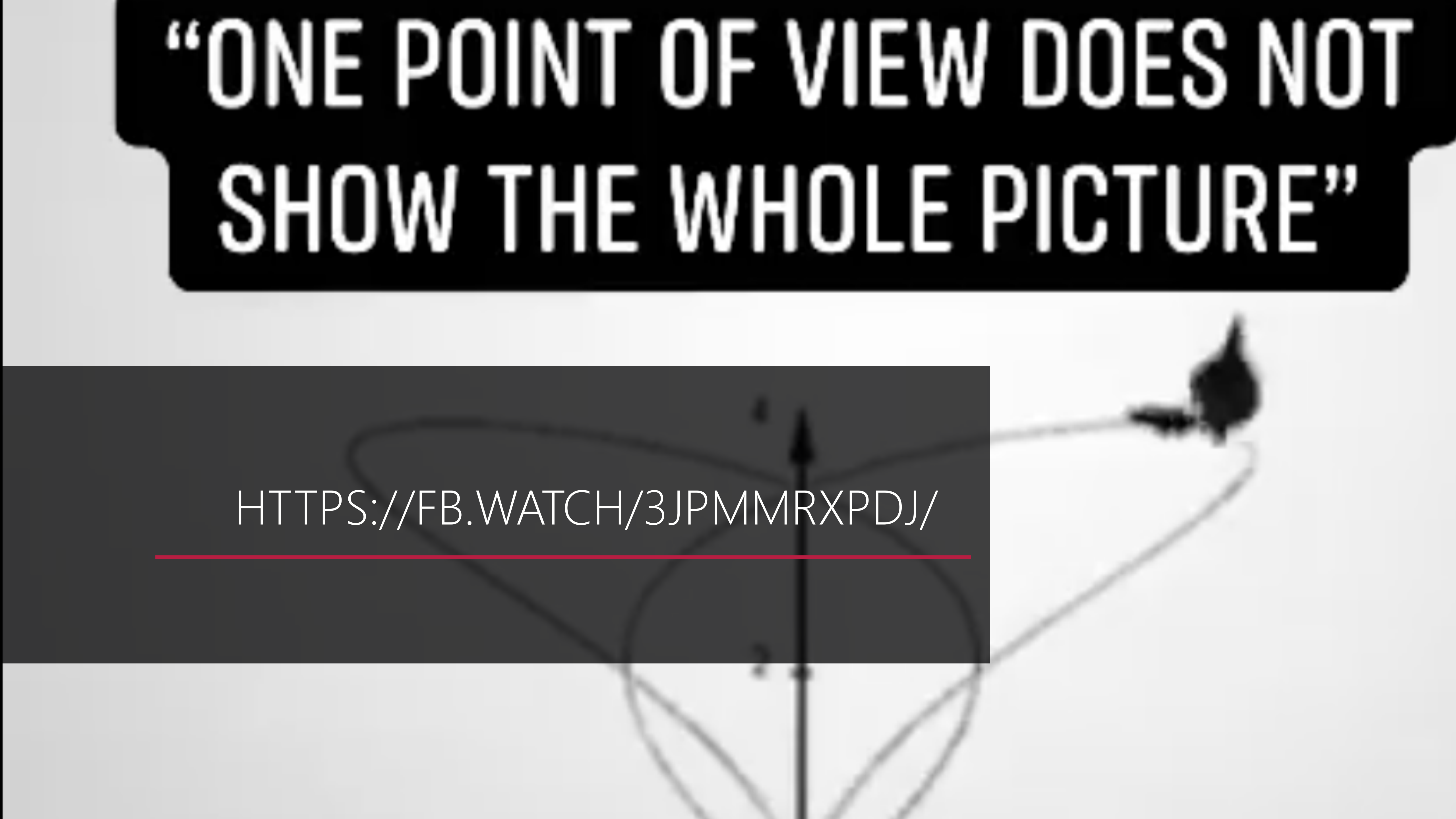
Available in genism python libarary:

- conceptnet-numberbatch-17-06-300 (1917247 records): ConceptNet Numberbatch consists of state...
- fasttext-wiki-news-subwords-300 (999999 records): 1 million word vectors trained on Wikipe...
- glove-twitter-100 (1193514 records): Pre-trained vectors based on 2B tweets,...
- glove-twitter-200 (1193514 records): Pre-trained vectors based on 2B tweets, ...
- glove-twitter-25 (1193514 records): Pre-trained vectors based on 2B tweets, ...
- glove-twitter-50 (1193514 records): Pre-trained vectors based on 2B tweets, ...
- glove-wiki-gigaword-100 (400000 records): Pre-trained vectors based on Wikipedia 2...
- glove-wiki-gigaword-200 (400000 records): Pre-trained vectors based on Wikipedia 2...
- glove-wiki-gigaword-300 (400000 records): Pre-trained vectors based on Wikipedia 2...
- glove-wiki-gigaword-50 (400000 records): Pre-trained vectors based on Wikipedia 2...
- word2vec-google-news-300 (3000000 records): Pre-trained vectors trained on a part of...
- word2vec-ruscorpora-300 (184973 records): Word2vec Continuous Skipgram vectors tra...

Vector Semantics
Vector Space
Transformation
Linear Algebra



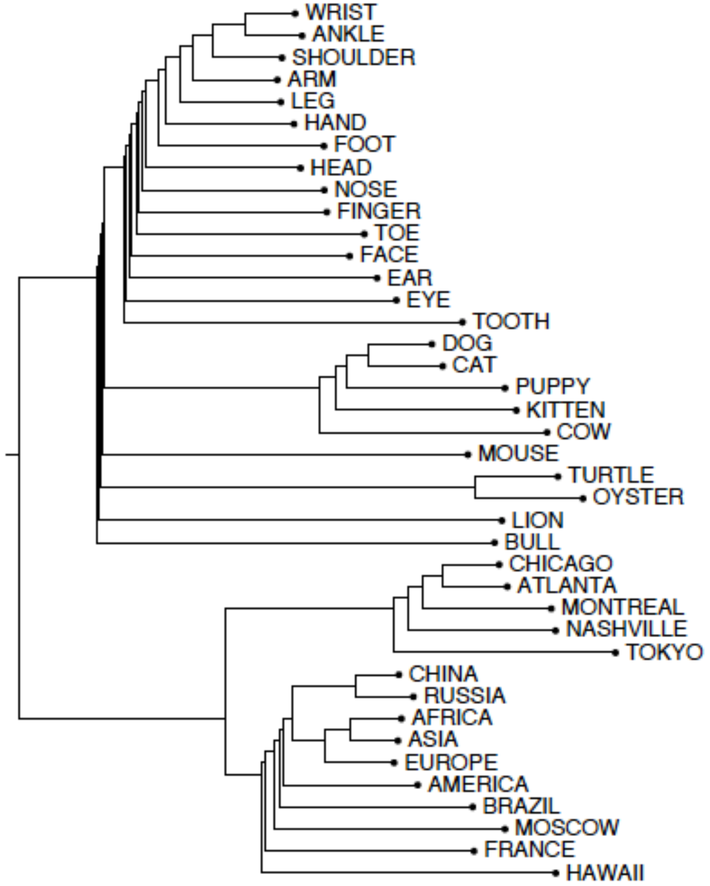
Hierarchy of mathematical spaces. Normed vector spaces are a superset of inner product spaces and a subset of metric spaces, which in turn is a subset of topological vector space.



HTTPS://FB.WATCH/3JPMMRXPDJ/

Visualization

Intuition, Geometry



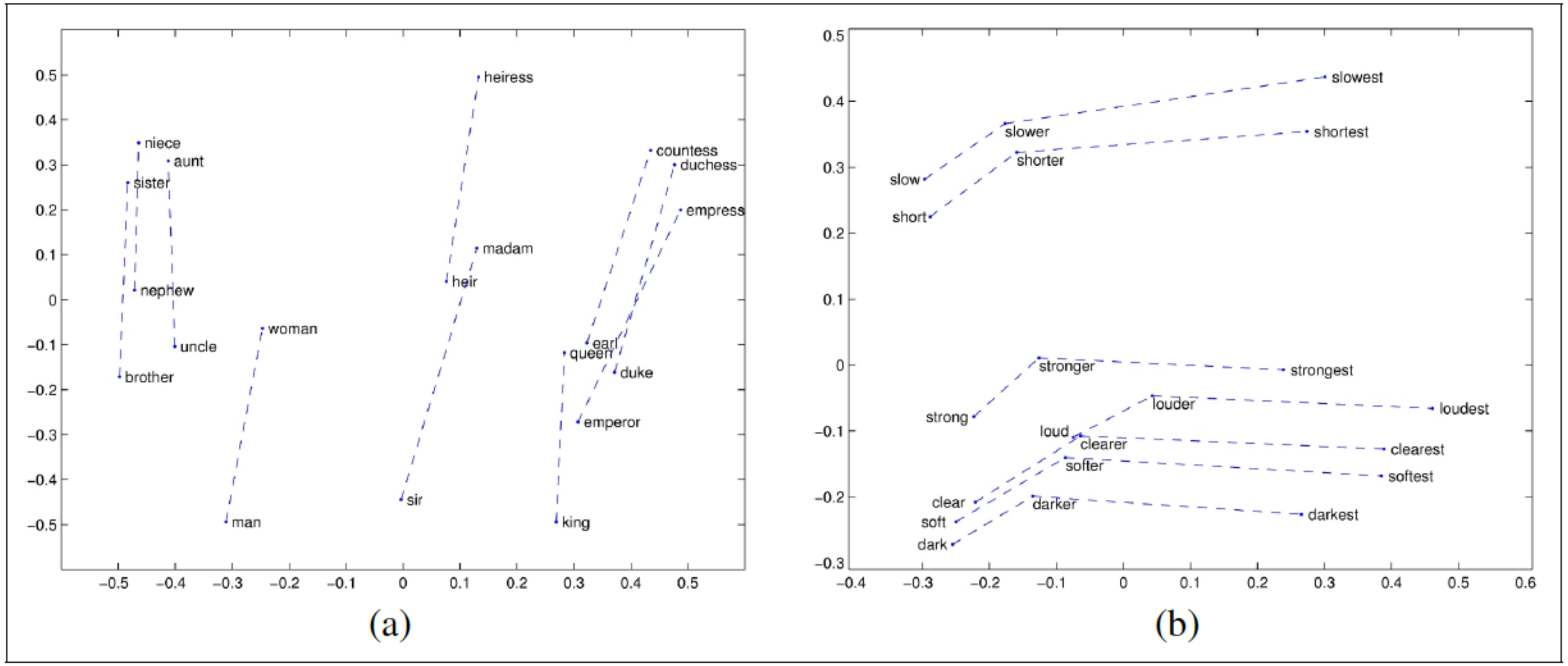


Figure 6.13 Relational properties of the vector space, shown by projecting vectors onto two dimensions. (a) 'king' - 'man' + 'woman' is close to 'queen' (b) offsets seem to capture comparative and superlative morphology (Pennington et al., 2014).

Movement

Temporality (How?)

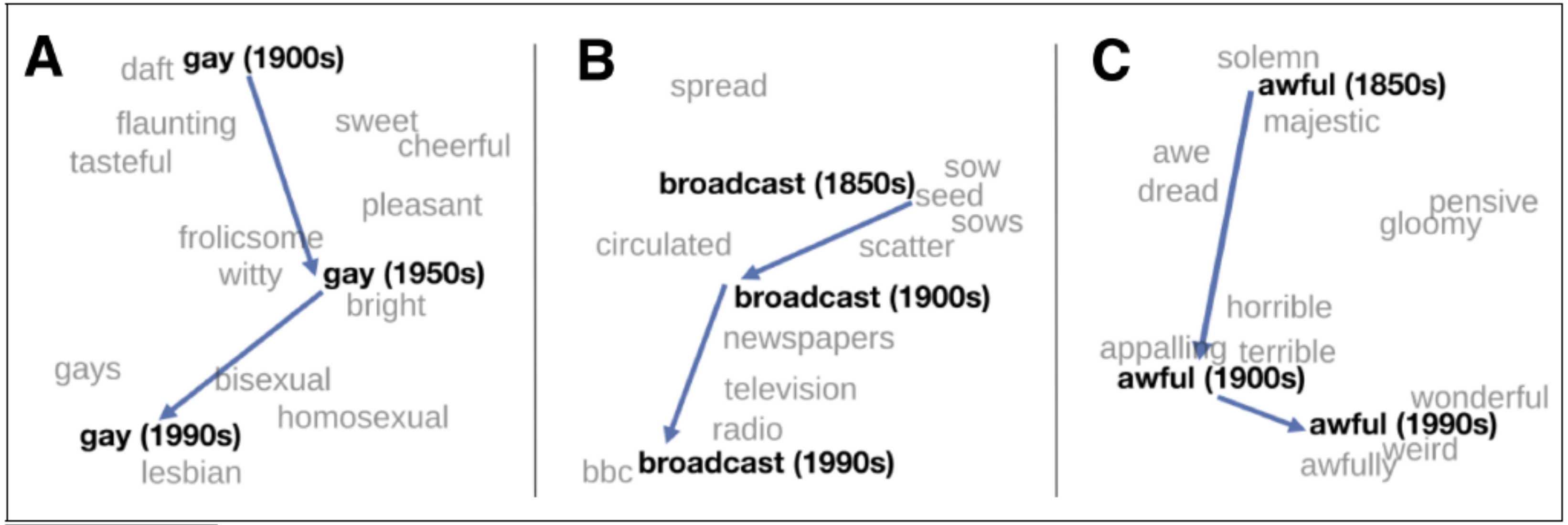


Figure 6.14 A t-SNE visualization of the semantic change of 3 words in English using word2vec vectors. The modern sense of each word, and the grey context words, are computed from the most recent (modern) time-point embedding space. Earlier points are computed from earlier historical embedding spaces. The visualizations show the changes in the word *gay* from meanings related to "cheerful" or "frolicsome" to referring to homosexuality, the development of the modern "transmission" sense of *broadcast* from its original sense of sowing seeds, and the pejoration of the word *awful* as it shifted from meaning "full of awe" to meaning "terrible or appalling" (Hamilton et al., 2016b).

Biases

Inherent/Latent/Hidden Distribution

- (sare, mom, nurse), (mr., ahmed, doctor, president)
- (drug, mexican), (education, usa, canada)
- (flowers, pleasant, {European-American}), (insects, ugly, {African-American})

Debiasing

- Gender-base: [he] remains masculine, [she] remains feminine, but [nurse],[doctor],[president] becomes neutral

Study of Bias in History

Evaluation

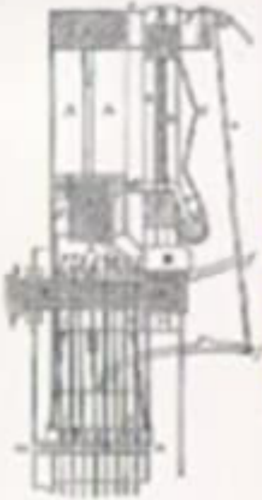
Intrinsic

- Golden Standards for Semantic Similarity/Distance
  - No Context: just pair of words
    - WordSim-353
    - SimLex-999
  - With Context:
    - Stanford Contextual Word Similarity (SCWS) (Huang et al., 2012) and the
    - Word-in-Context (WiC) (Pilehvar and Camacho-Collados, 2019)

Extrinsic:

- Improve the performance of underlying task
  - Information Retrieval (IR), Document Classification, Sentiment Analysis, …

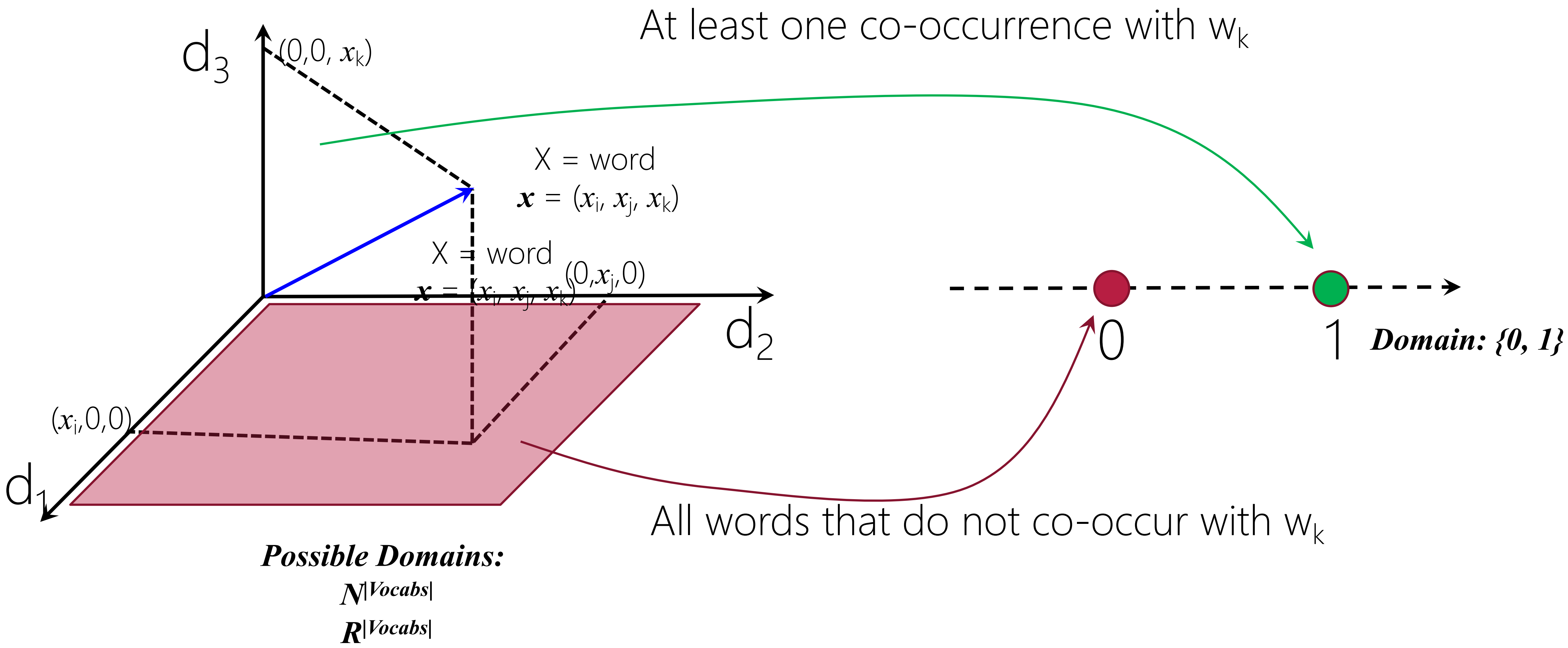
How to learn representation for sentence/paragraph/documents?

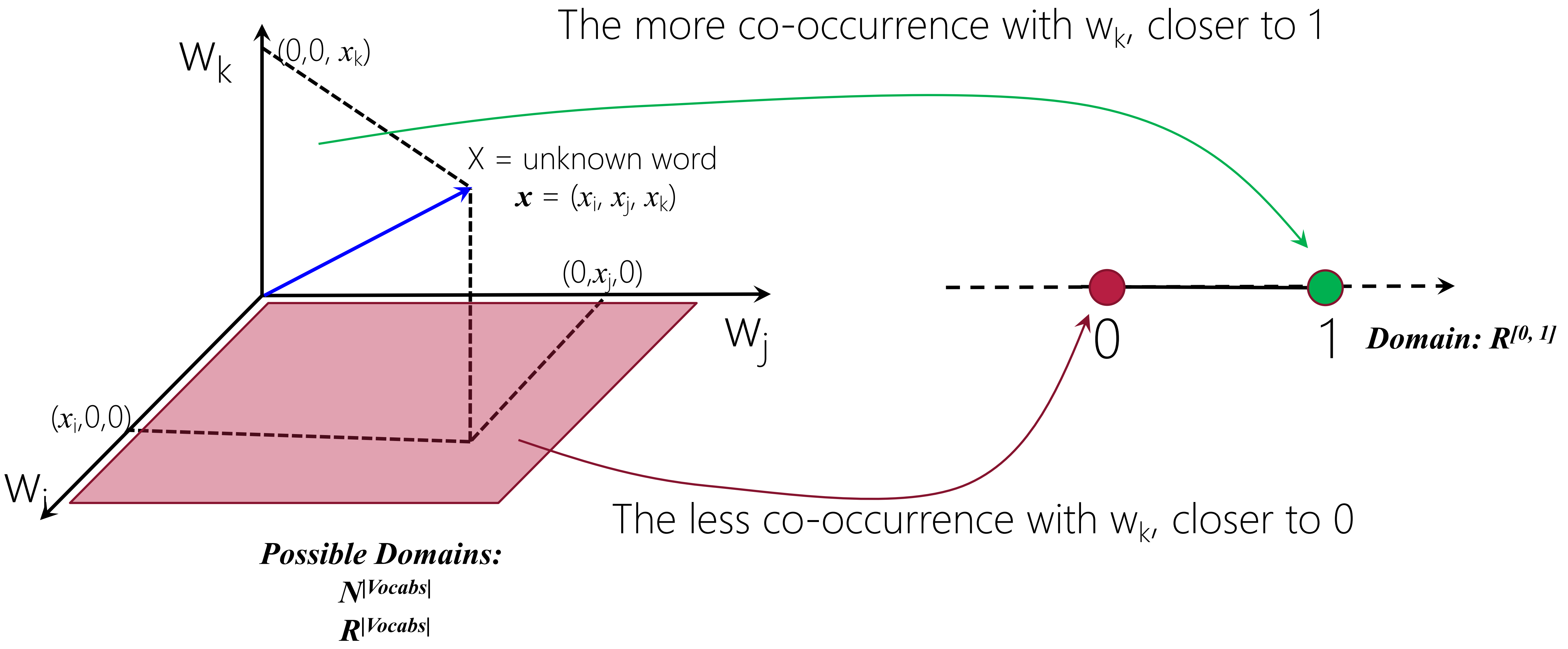


SPEECH and LANGUAGE PROCESSING

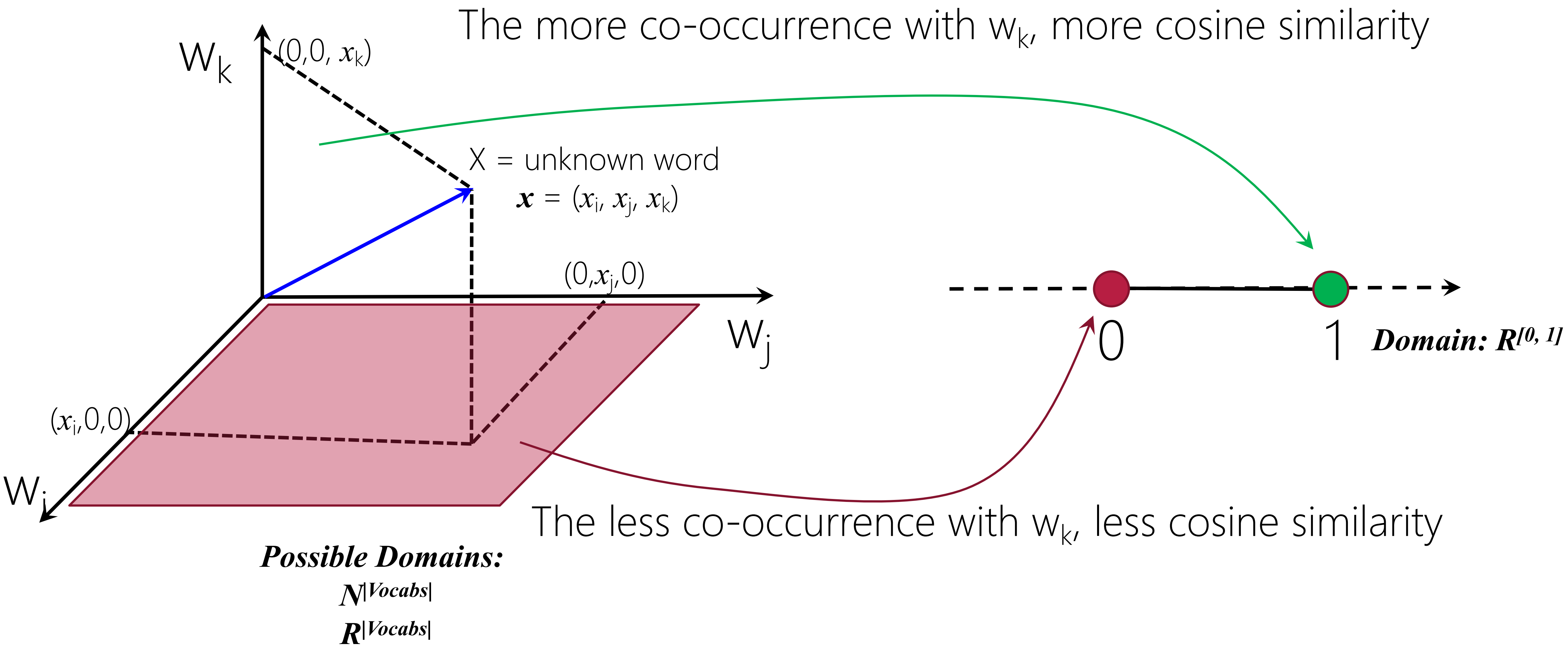
An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition

DANIEL JURAFSKY & JAMES H. MARTIN


At least one co-occurrence with w_k
d_3
$(0,0, x_k)$
X = word
$\boldsymbol{x} = (x_i, x_j, x_k)$
X = word
$\boldsymbol{x} = (x_i, x_j, x_k)$
$(0,x_j,0)$
d_2
$(x_i,0,0)$
d_1
0
1
Domain: {0, 1}
All words that do not co-occur with w_k
Possible Domains:
$N^{|Vocabs|}$
$R^{|Vocabs|}$



The more co-occurrence with w_k, closer to 1
W_k
$(0,0, x_k)$
X = unknown word
$\boldsymbol{x} = (x_i, x_j, x_k)$
$(0,x_j,0)$
W_j
$(x_i,0,0)$
W_i
0
1
Domain: $R^{[0, 1]}$
The less co-occurrence with w_k, closer to 0
Possible Domains:
$N^{|Vocabs|}$
$R^{|Vocabs|}$



The more co-occurrence with w_k, more cosine similarity
W_k
$(0,0, x_k)$
X = unknown word
$\boldsymbol{x} = (x_i, x_j, x_k)$
$(0,x_j,0)$
W_j
$(x_i,0,0)$
W_i
0
1
Domain: $R^{[0, 1]}$
The less co-occurrence with w_k, less cosine similarity
Possible Domains:
$N^{|Vocabs|}$
$R^{|Vocabs|}$