# FEATURE SPACES III

Prof. Alexander Huth
10/19/2017

# HOMEWORKS

* Homework 1 will be graded/returned by next Thursday (10/26)

* Homework 2 out next Thursday (10/26)

# SYSTEM IDENTIFICATION

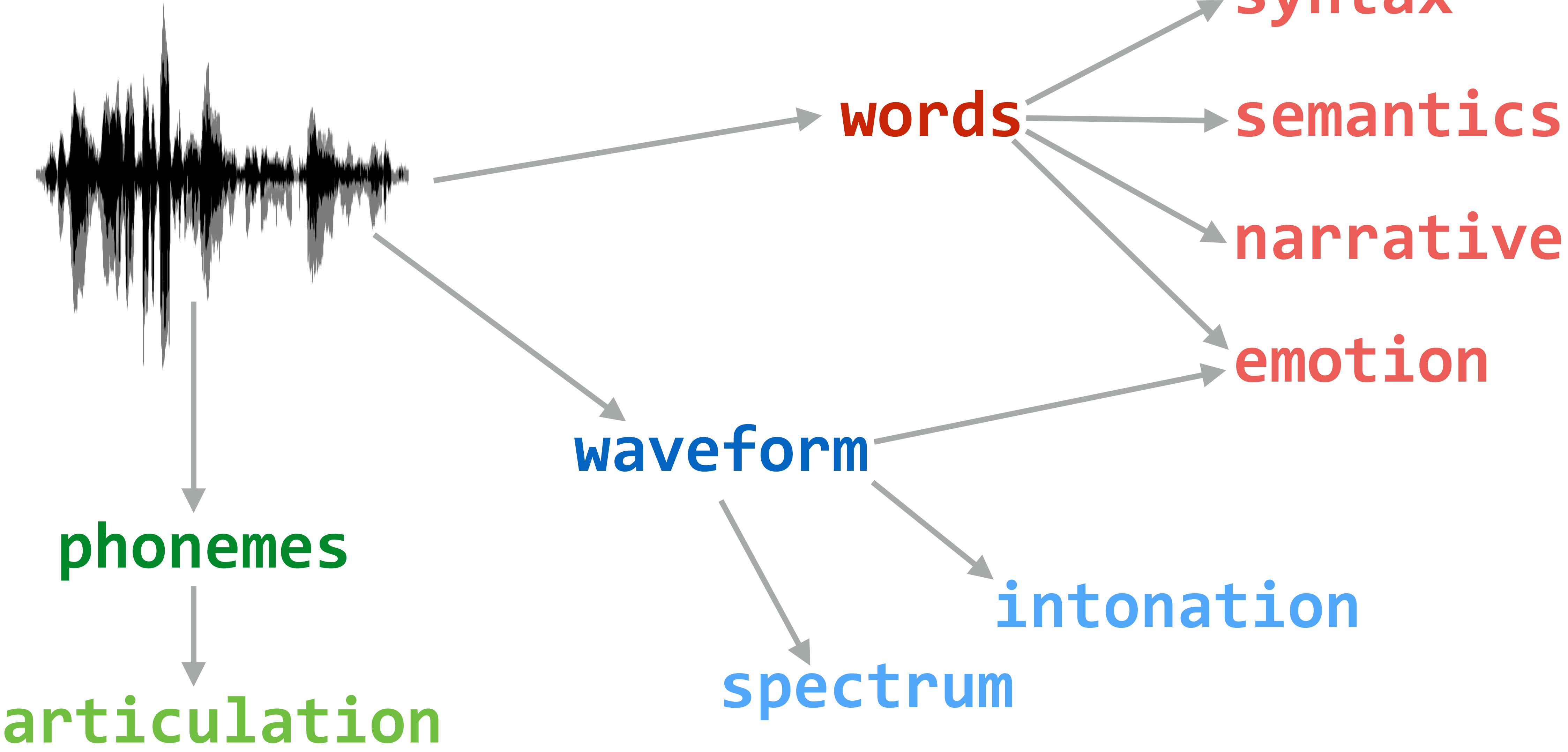
* Linearized model

$$Y = \mathbb{L}(X)\beta$$

Let’s invent some **L**’s

# LANGUAGE

*“Now this is a story all about how my life got flipped-turned upside down…”*



words

syntax

semantics

narrative

emotion

waveform

intonation

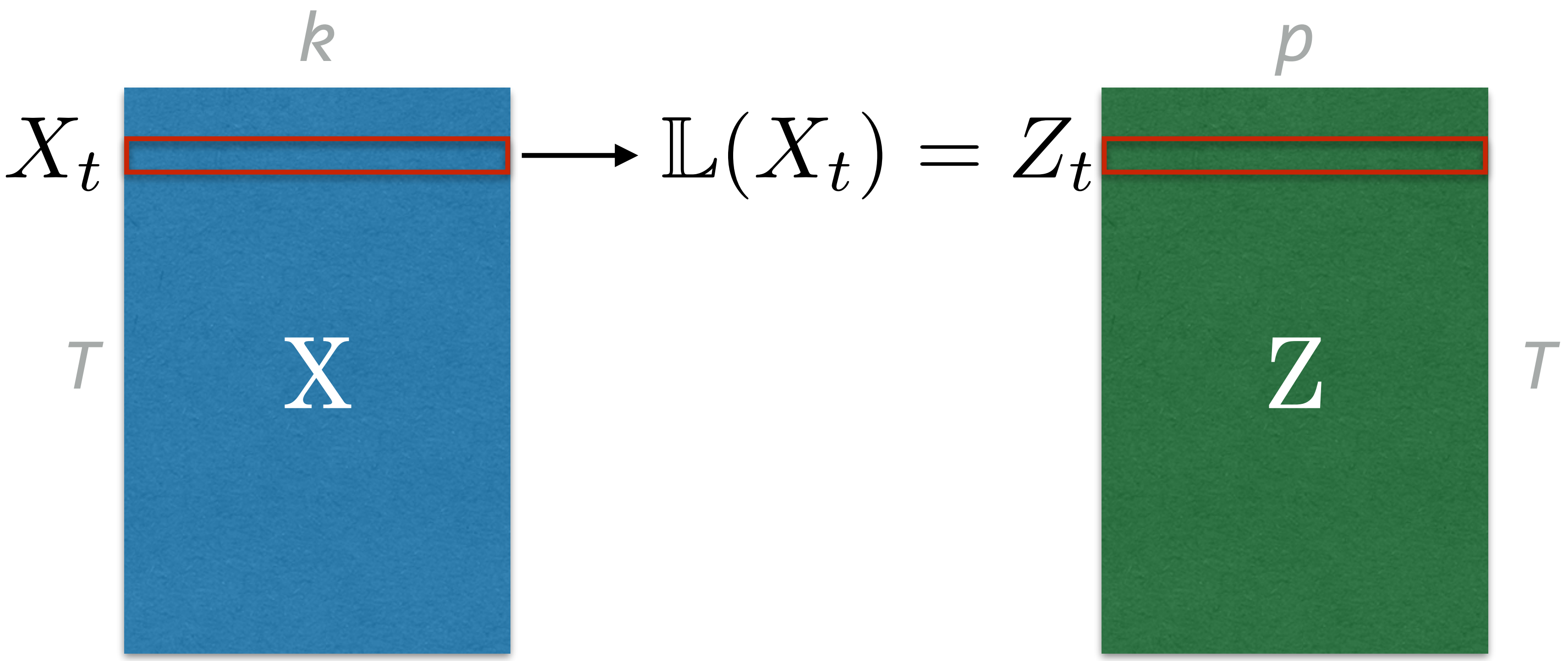
spectrum

phonemes

articulation

# LINEARIZING TRANSFORMATIONS

* Simplest version: time-invariant **L**



$X_t \longrightarrow \mathbb{L}(X_t) = Z_t$

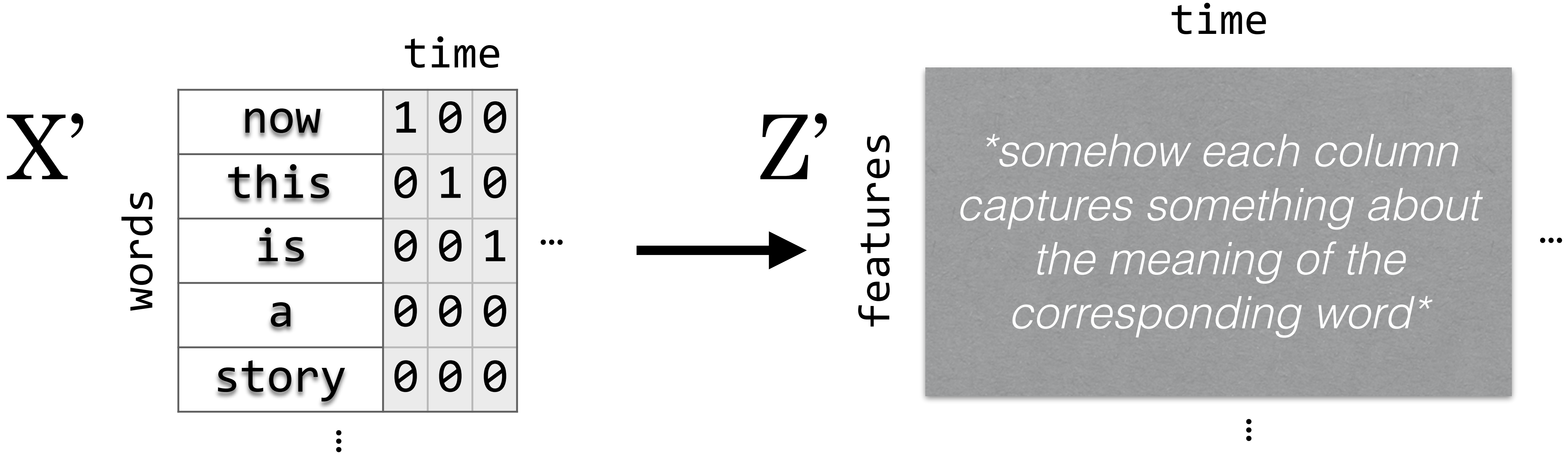
# LEXICAL SEMANTICS

- Let’s create an **L** that captures word-level semantic information

- Unlike the awful syntax models, this model will be *time-invariant*

# LEXICAL SEMANTICS

$$\mathbb{L}_{semantic}(\overset{\substack{\text{1-hot vector}\\ \downarrow}}{X_t}) = \overset{\substack{\text{p-dim vector}\\ \downarrow}}{Z_t} \in \mathbb{R}^p$$



X'

| words | time | | |
|---|---|---|---|
| now | 1 | 0 | 0 |
| this | 0 | 1 | 0 |
| is | 0 | 0 | 1 |
| a | 0 | 0 | 0 |
| story | 0 | 0 | 0 |

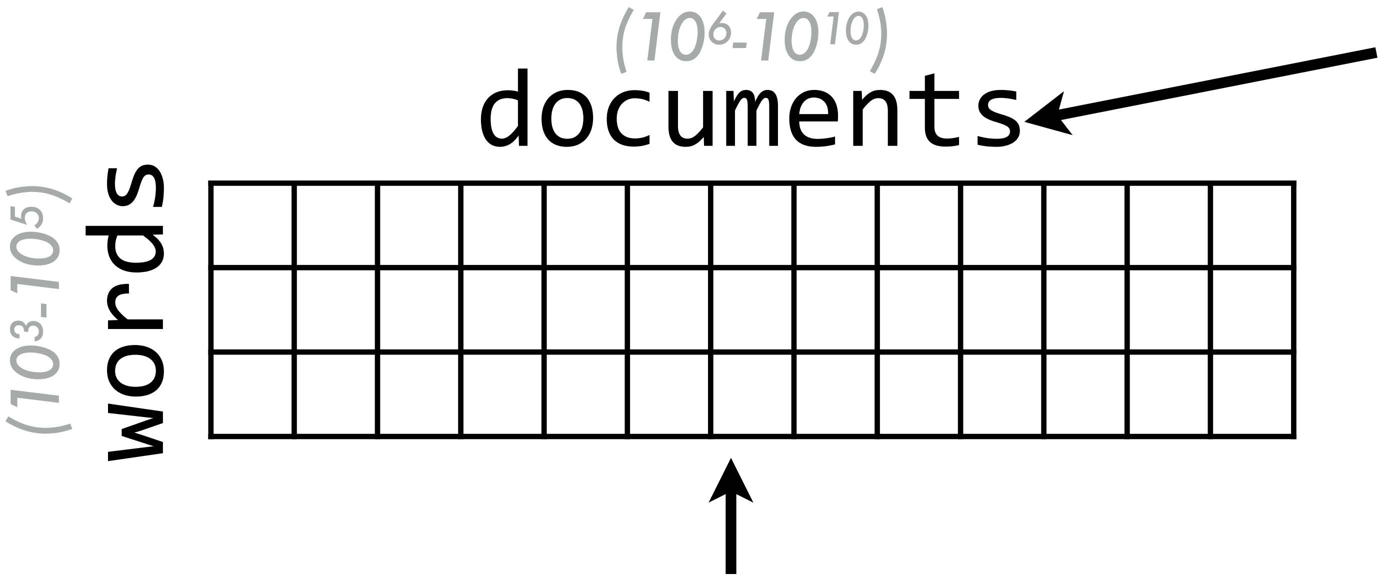
⋮ …

Z'

**somehow each column captures something about the meaning of the corresponding word**

# LEXICAL SEM. - LSA

* Latent Semantic Analysis (LSA)



($10^6$-$10^{10}$)

documents

($10^3$-$10^5$)

words

drawn from a large training corpus

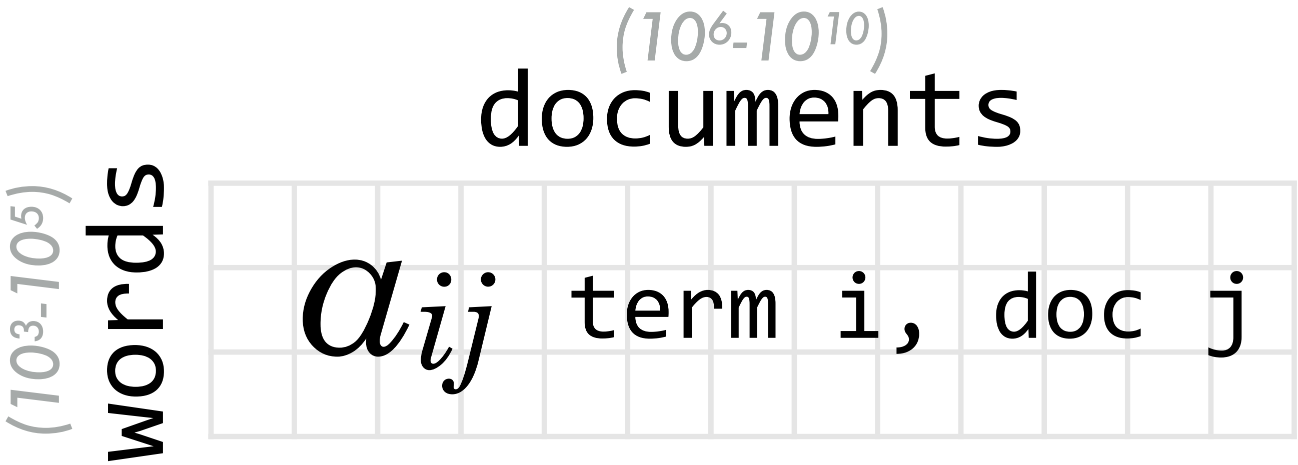
(split up large docs!)

*extremely* sparse matrix (0.5% nonzero)

Deerwester et al. (1988)

# LEXICAL SEM. - LSA

- Latent Semantic Analysis (LSA)



$(10^6\text{-}10^{10})$

documents

$(10^3\text{-}10^5)$

words

$a_{ij}$ term i, doc j

Often the entries are normalized

$$a_{ij} = \frac{\text{tf}_{ij}}{\log_2 \frac{n}{1+\text{df}_i}}$$

# word $i$ in doc $j$

# docs

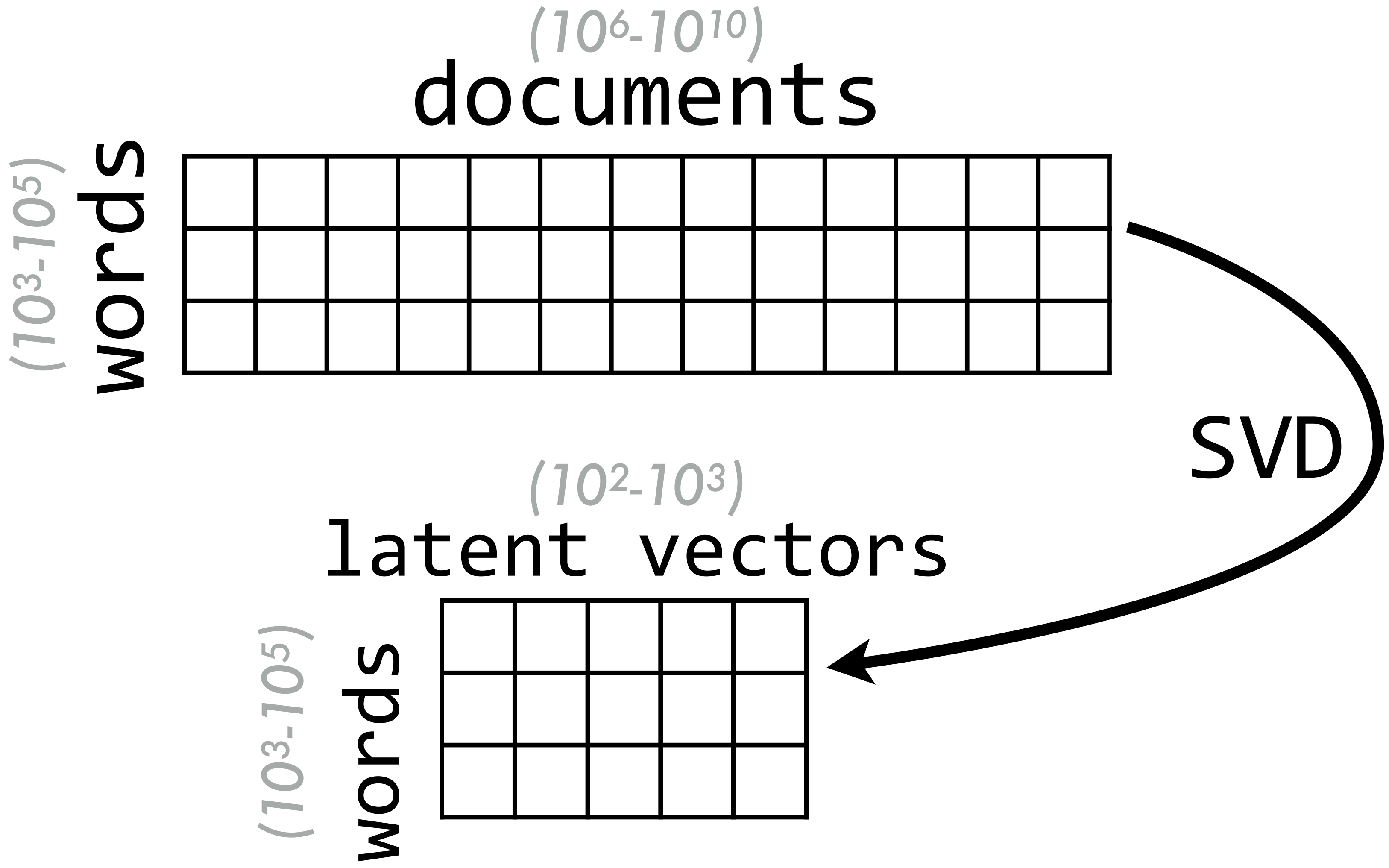
# docs with word $i$

“tf-idf”

{ij}}{\log_2
}

# LEXICAL SEM. - LSA

* Latent Semantic Analysis (LSA)



($10^6$-$10^{10}$)

documents

($10^3$-$10^5$) words

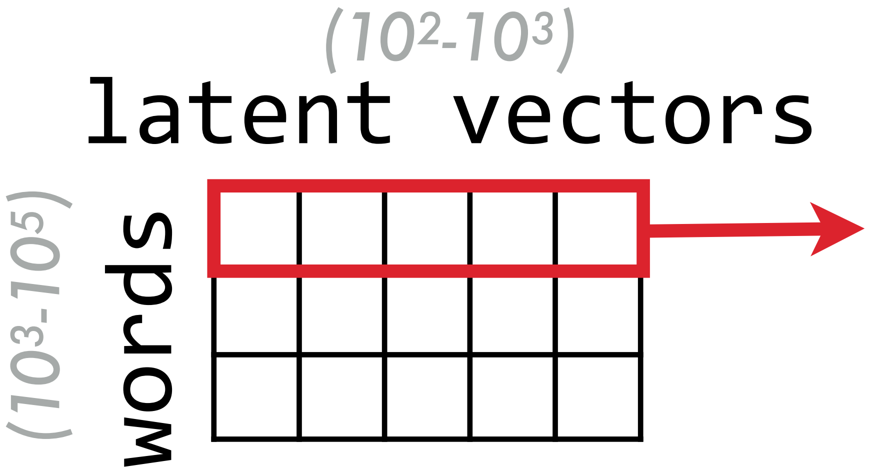
SVD

($10^2$-$10^3$)

latent vectors

($10^3$-$10^5$) words

# LEXICAL SEM. - LSA

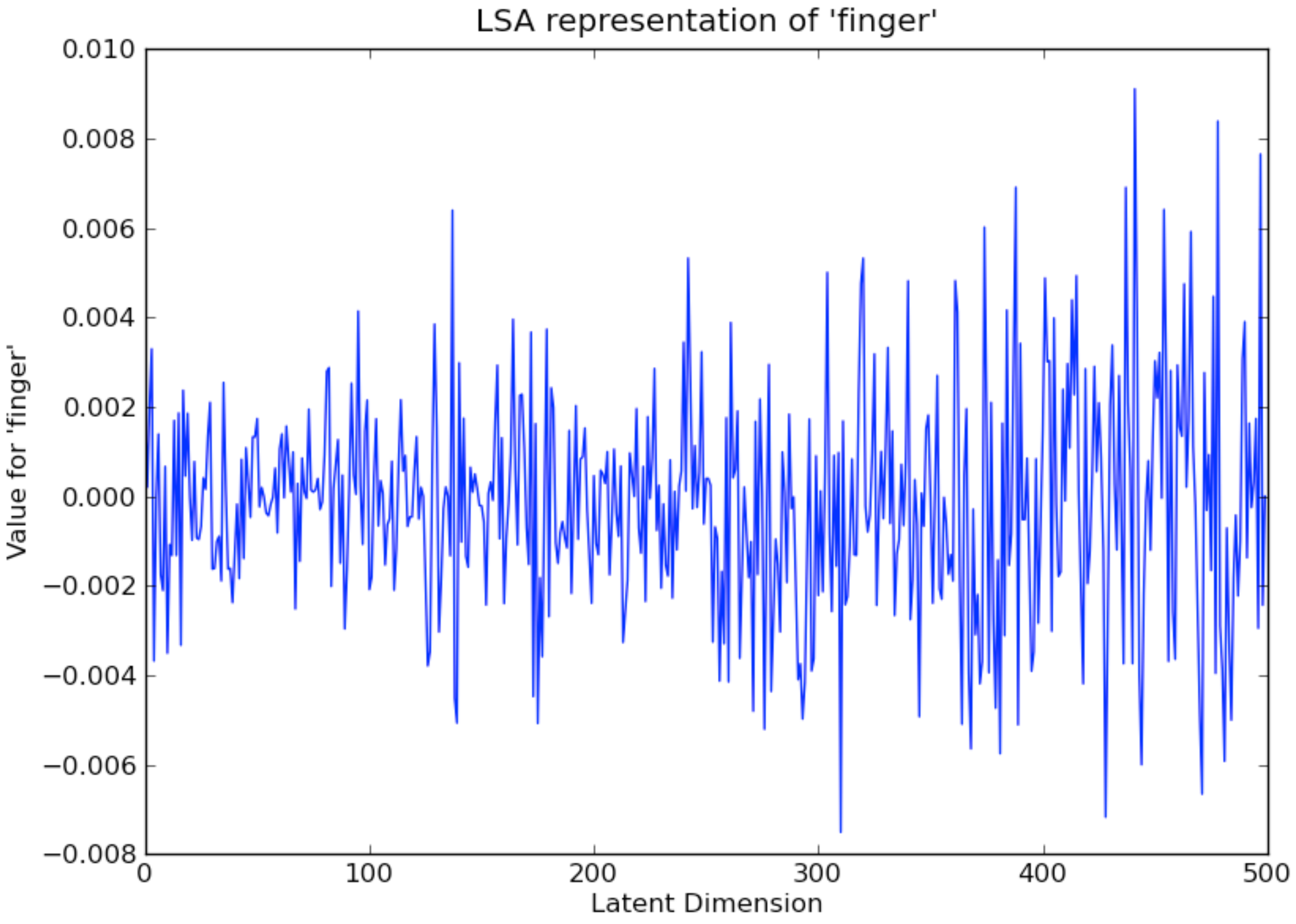


$(10^2\text{-}10^3)$

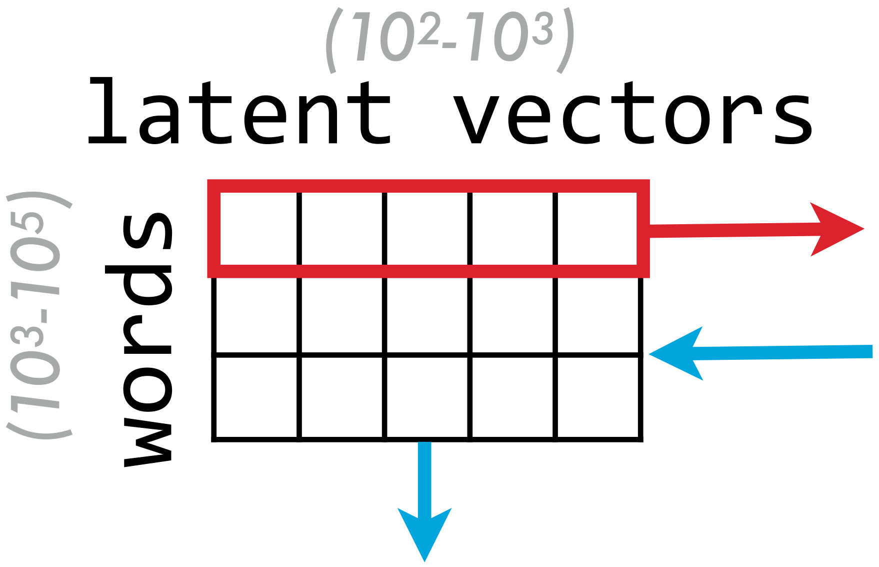
latent vectors

$(10^3\text{-}10^5)$

words



# LEXICAL SEM. - LSA



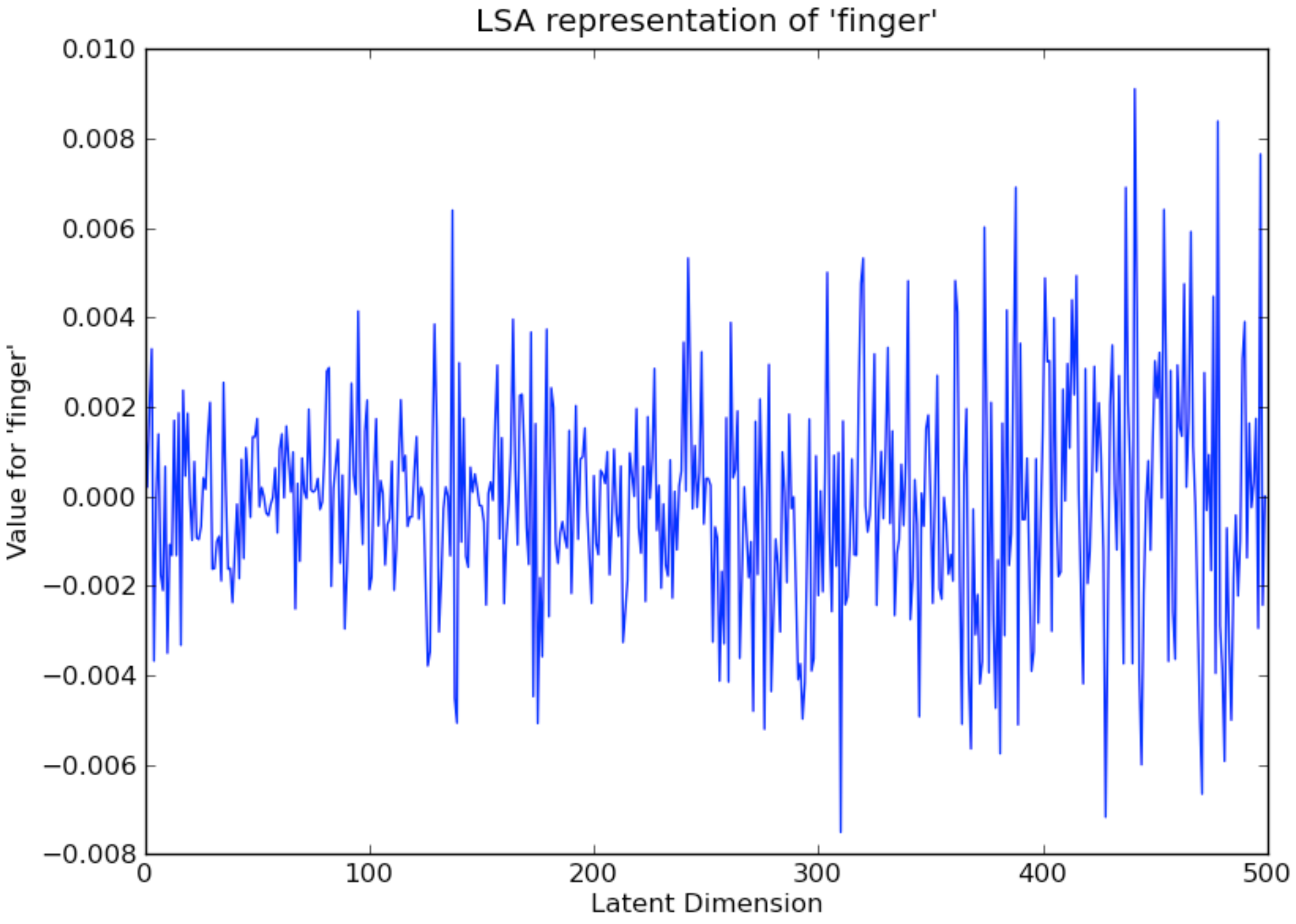
$(10^2$-$10^3)$

latent vectors

$(10^3$-$10^5)$

words

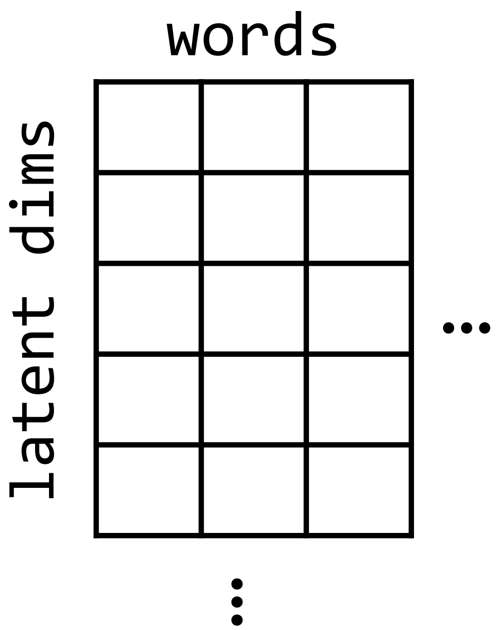
0.61- thumb
0.52- fingers
0.51- forefinger
0.49- tip
0.47- wrist
0.41- elbow
0.41- throat
0.41- lips
0.40- string

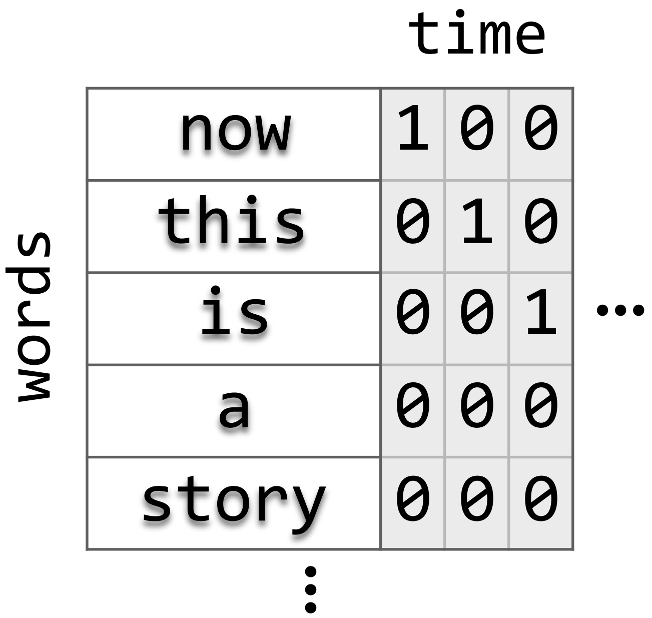


# LEXICAL SEM. - LSA

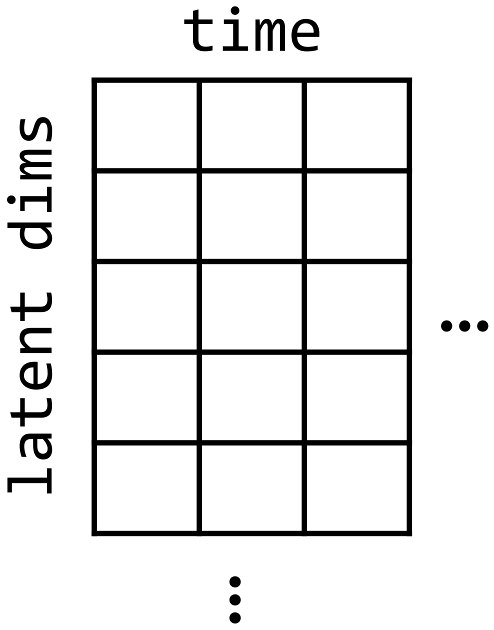
$$E \quad \cdot \quad X' \quad = \quad Z'$$

embedding matrix' * word matrix' = semantic stimulus matrix'





| | time | | |
|---|---|---|---|
| now | 1 | 0 | 0 |
| this | 0 | 1 | 0 |
| is | 0 | 0 | 1 |
| a | 0 | 0 | 0 |
| story | 0 | 0 | 0 |



## REMINDER FROM A FEW WEEKS AGO…

# TIKHONOV REGRESSION

- this is equivalent to **TIKHONOV REGRESSION** on the **WORDS** with a prior determined by the **WORD EMBEDDING**

$$\frac{1}{\sigma^2}\Sigma_\beta = (C^T C)^{-1} = E^T E$$

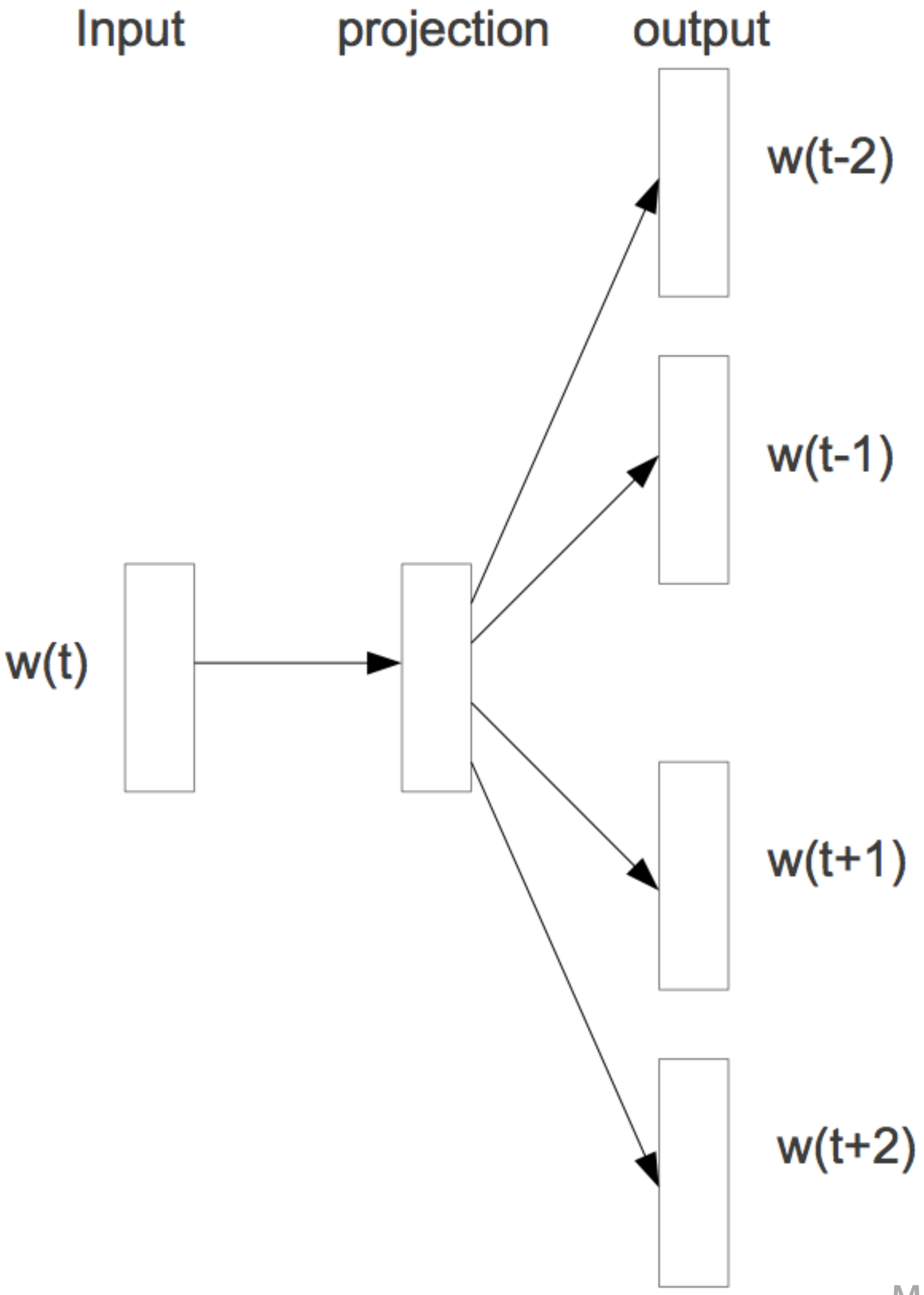
PRIOR COVARIANCE

INVERSE OF PENALTY INNER PRODUCT

EMBEDDING INNER PRODUCT

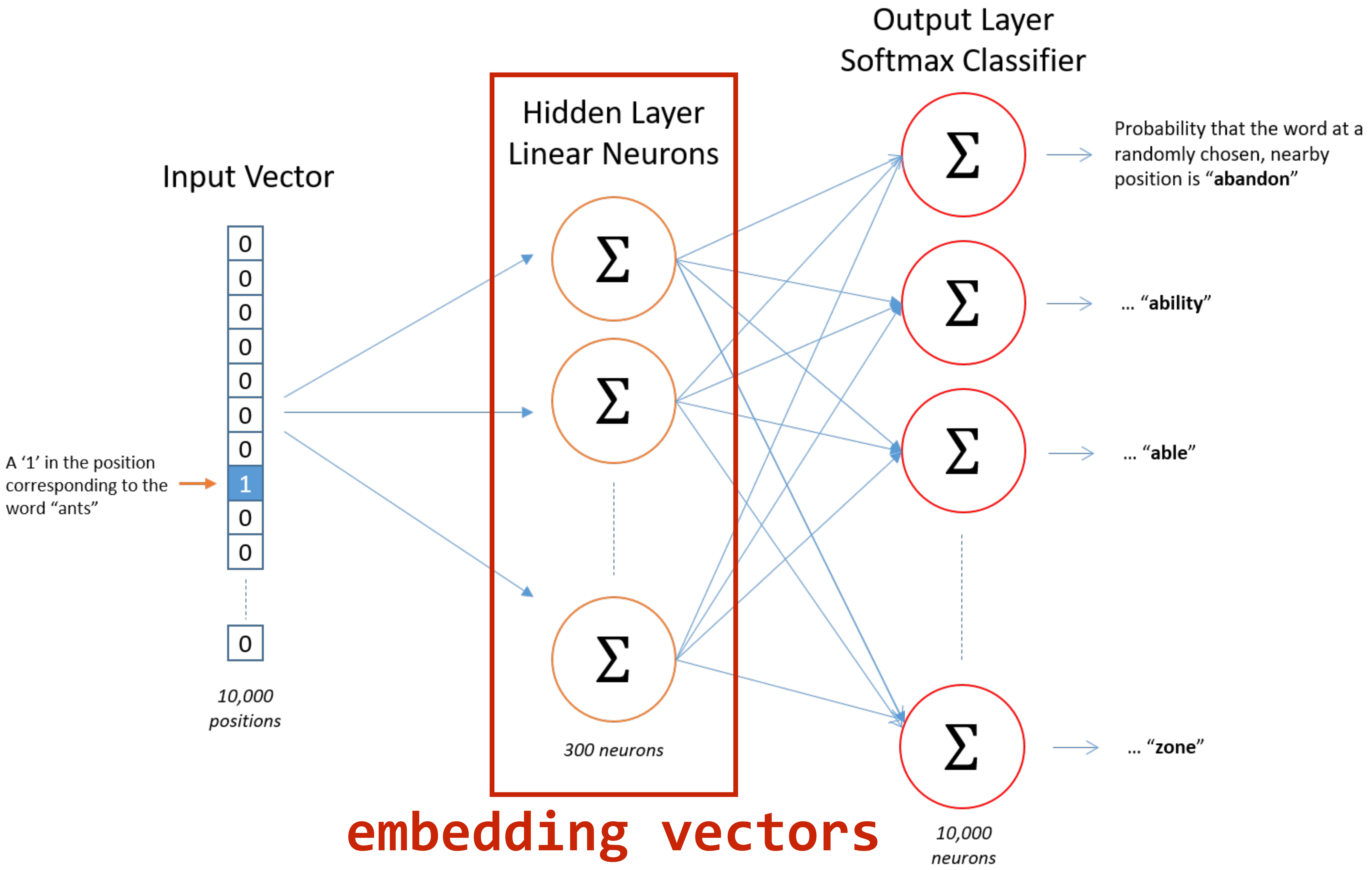
- **i.e. the prior covariance between two words’ weights is equal to the dot product of their embedding vectors**

# SEMANTICS - WORD2VEC



Mikolov et al. (2013)

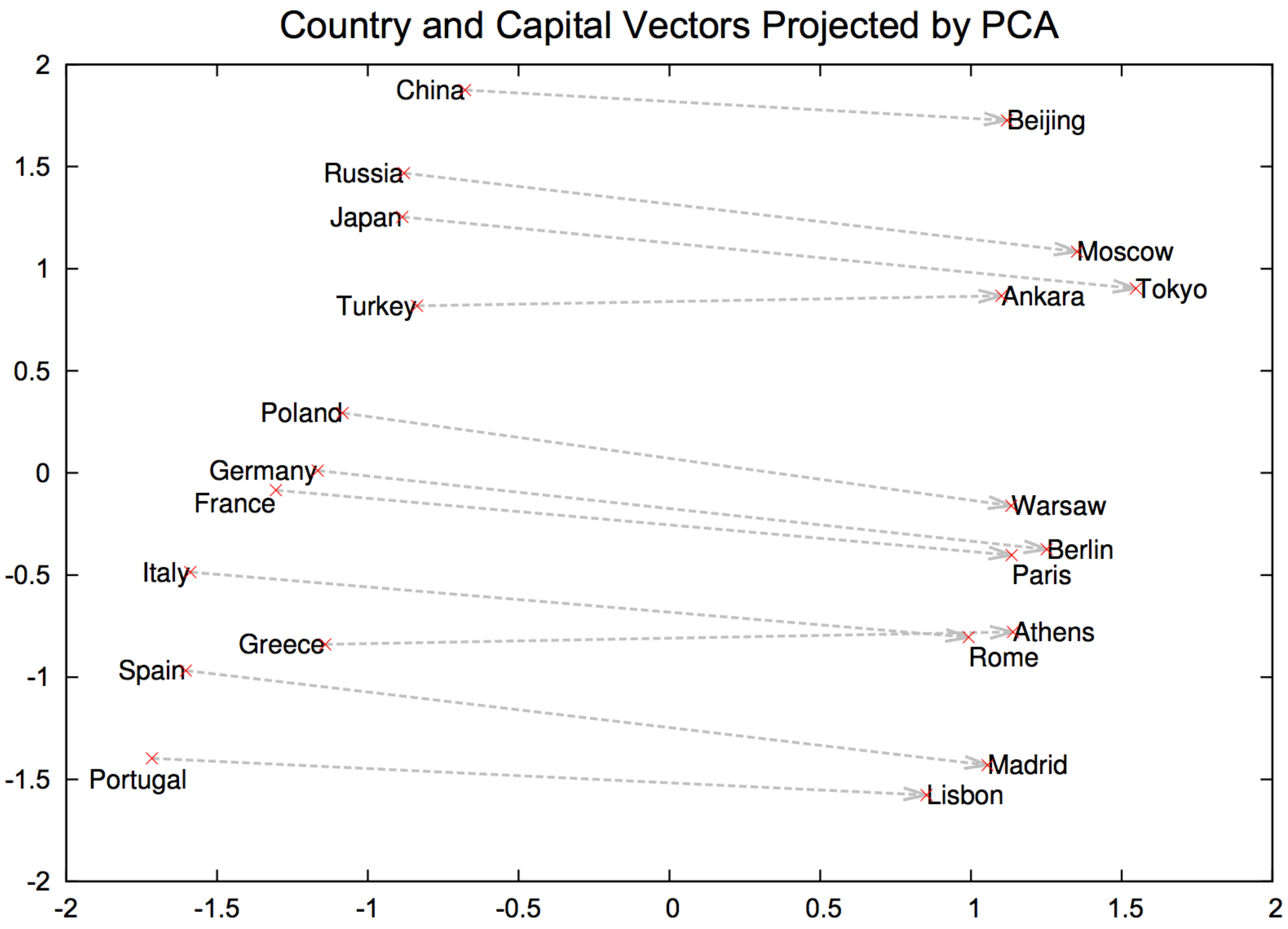
# SEMANTICS - WORD2VEC



embedding vectors

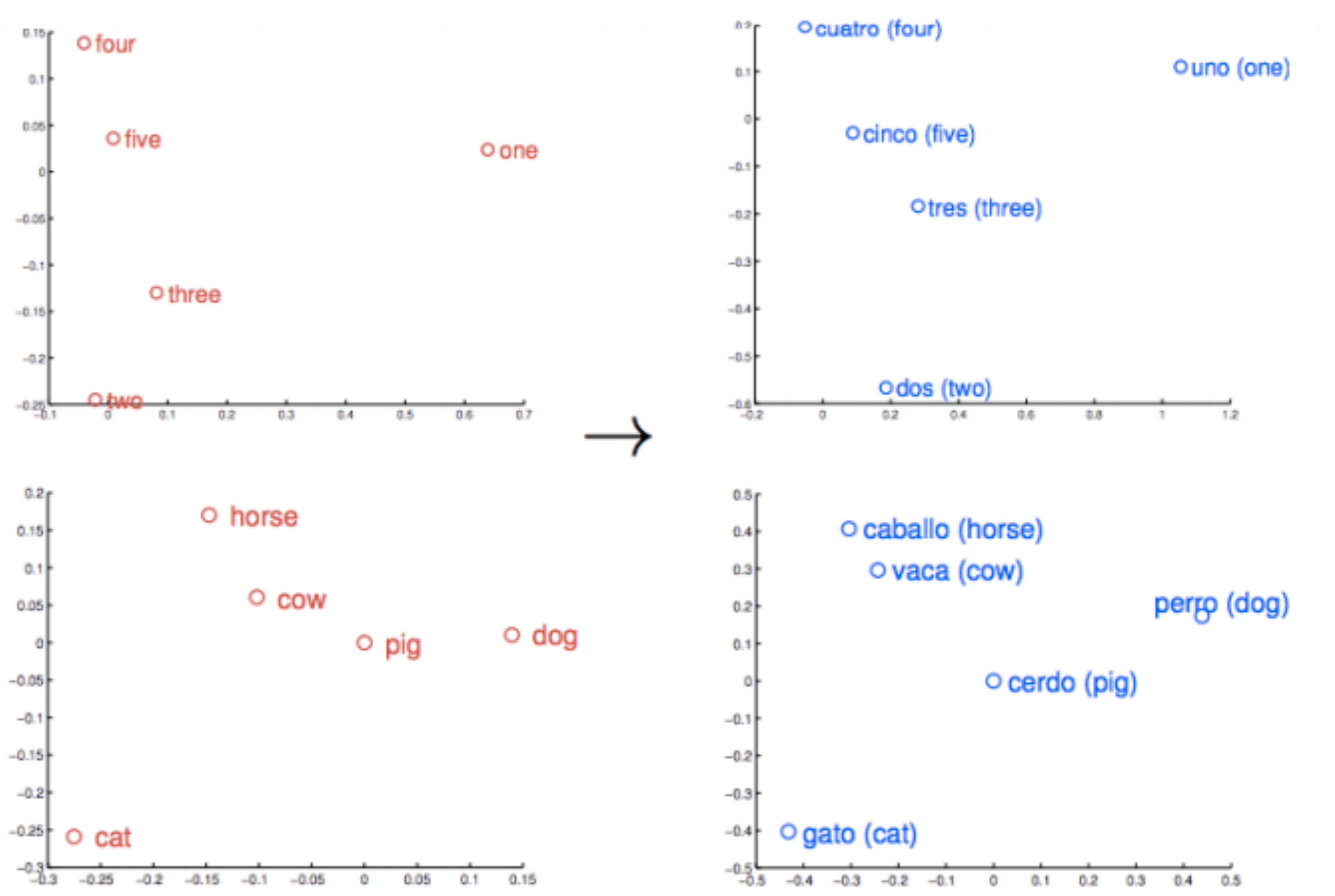
http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

# SEMANTICS - WORD2VEC



Mikolov et al. (2013)

# SEMANTICS - WORD2VEC



https://deeplearning4j.org/word2vec

# SEMANTICS - ENGLISH-1000

…

difficult

…

husband

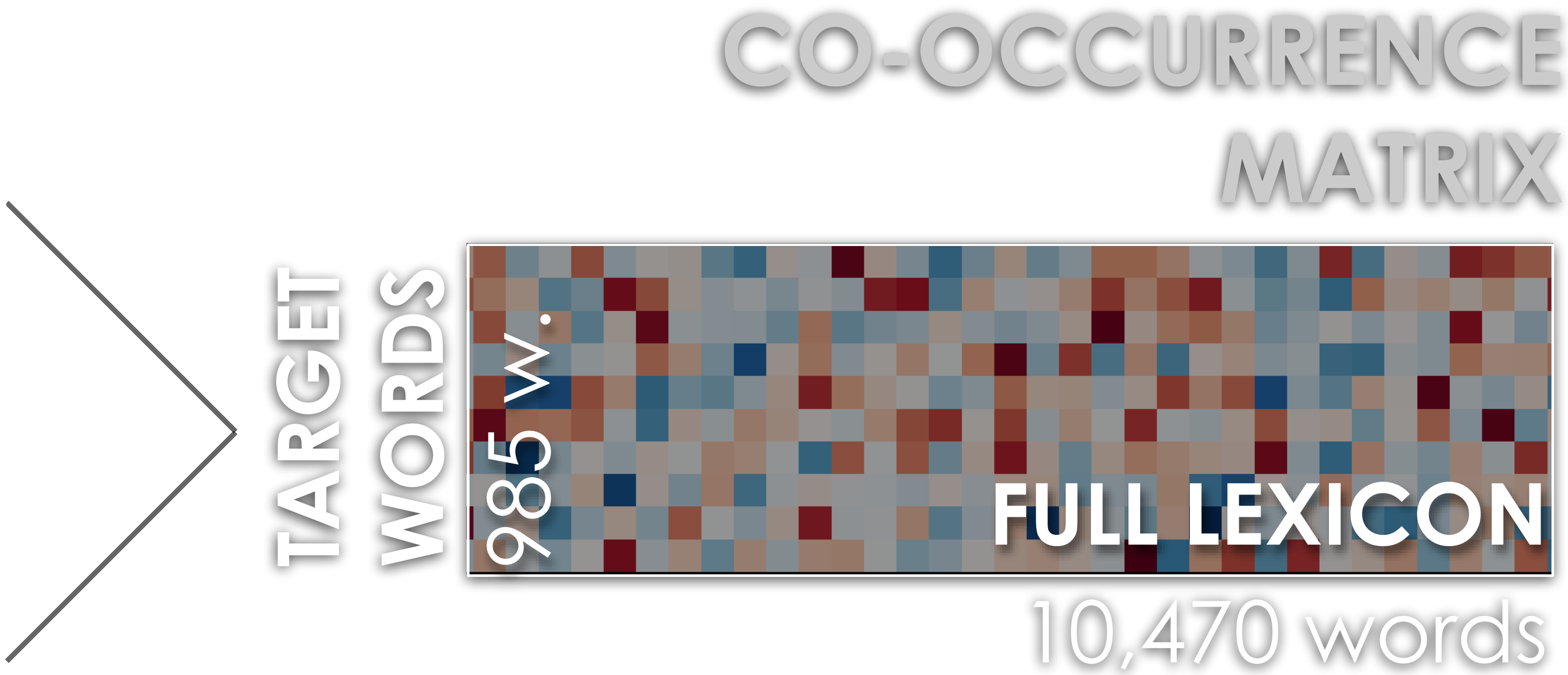
…

potato

…

remember

…



# SEMANTICS - ENGLISH-1000

* The corpus was used to build a 985 x 10,470 matrix *M*

  * $M_{i,j}$ is the number of times target *i* occurs within 15 words of word *j*

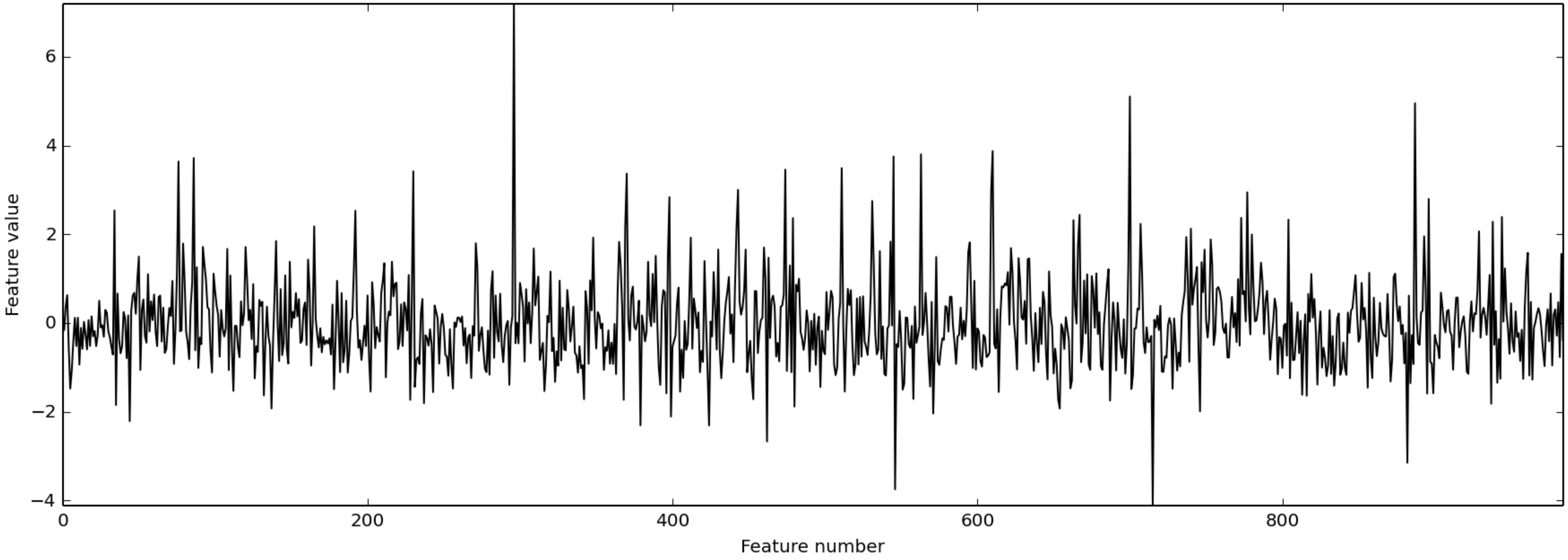
* Then log-transform: $M^{*}_{i,j} = \log(M_{i,j}+1)$

* Then *z*-score each row, then each column

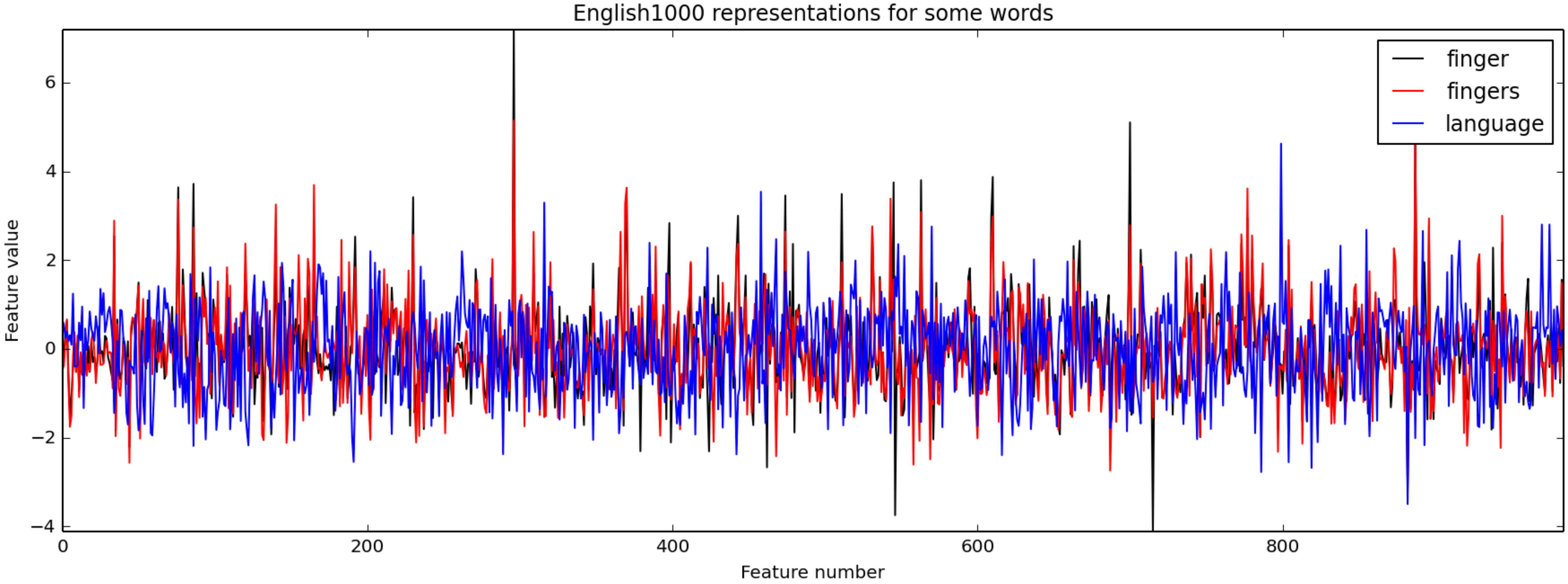
* ... yielding 985-D vector representation of each word in the lexicon

# SEMANTICS - ENGLISH-1000

“finger” in english-1000



# SEMANTICS - ENGLISH-1000



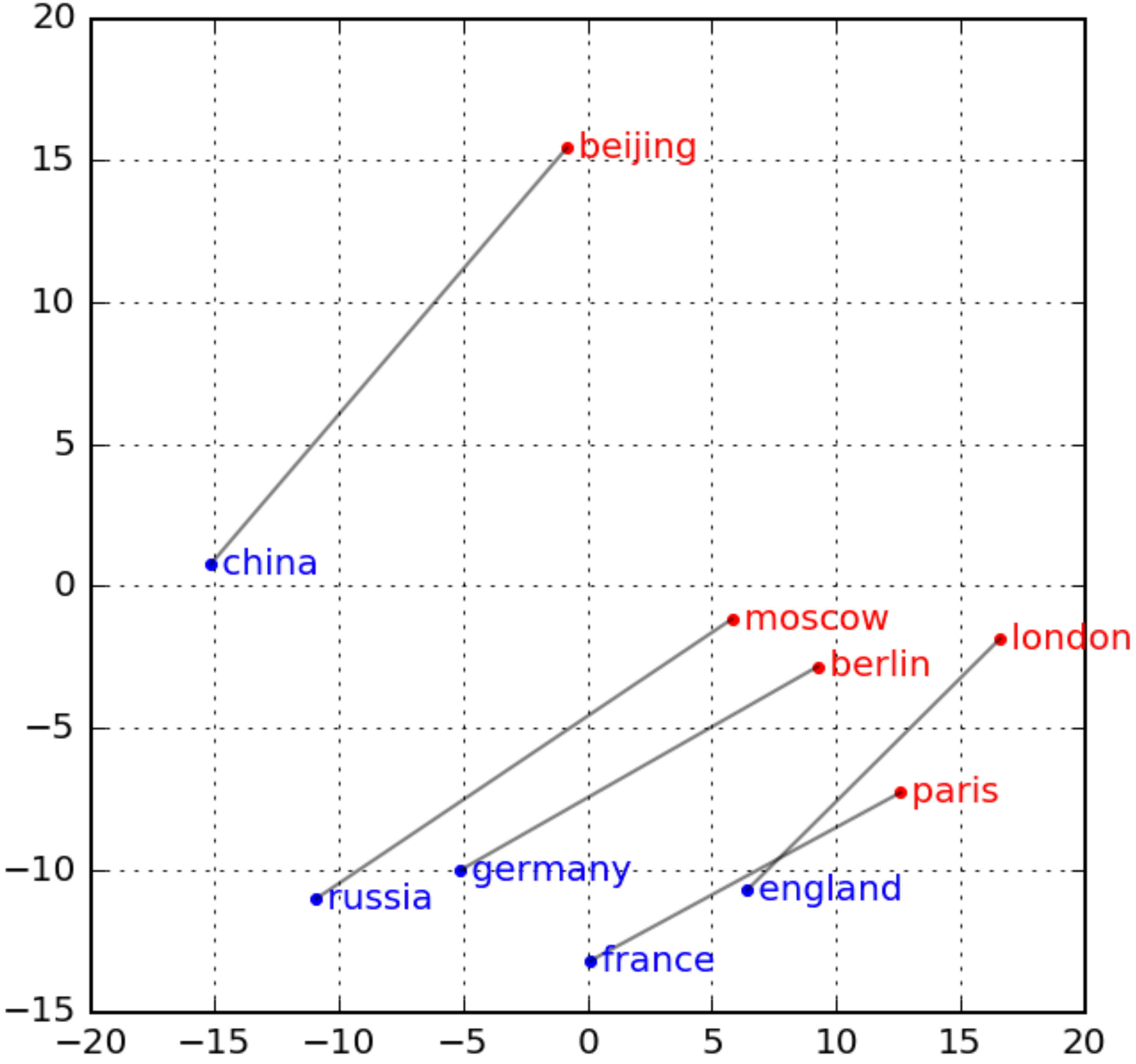
# SEMANTICS - ENGLISH-1000

correlation
with “finger”

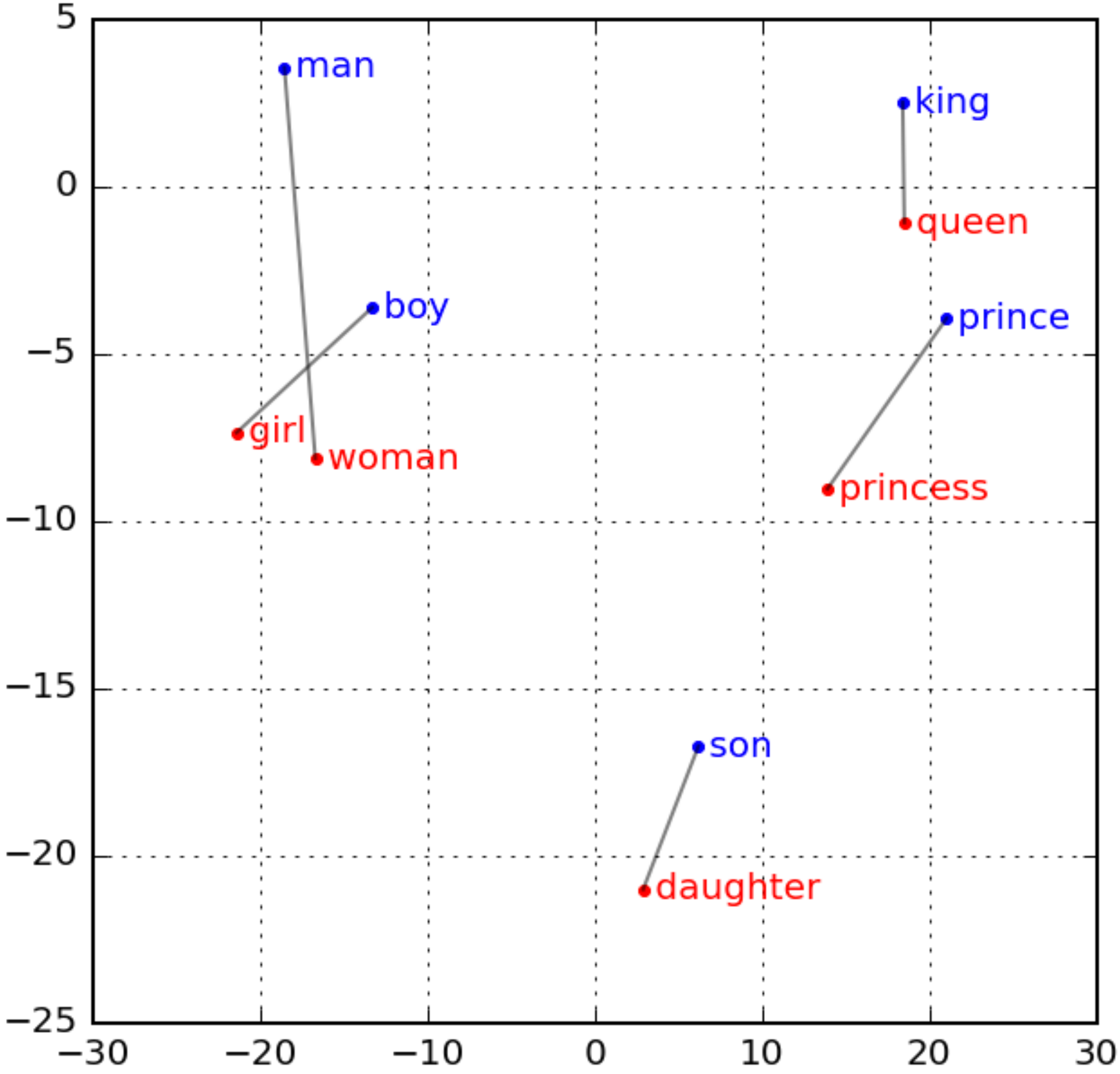
word

```
1.00, 'finger'
0.81, 'fingers'
0.67, 'hand'
0.67, 'nose'
0.66, 'arm'
0.64, 'mouth'
0.64, 'stick'
0.63, 'neck'
0.63, 'forehead'
0.62, 'tongue'
```

# SEMANTICS - ENGLISH-1000

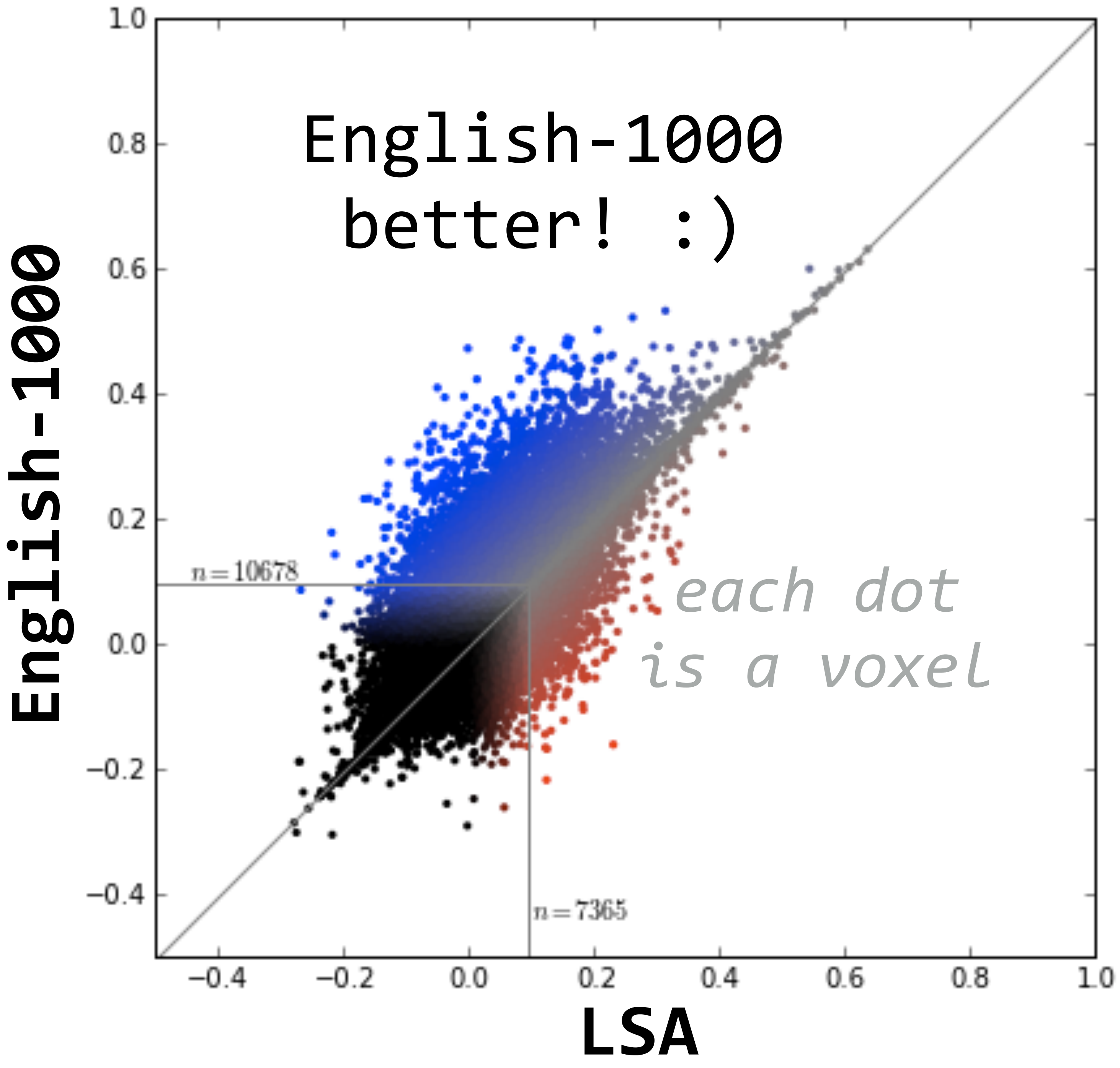


# SEMANTICS - ENGLISH-1000



# ENGLISH-1000 VS LSA

model performance on held-out data

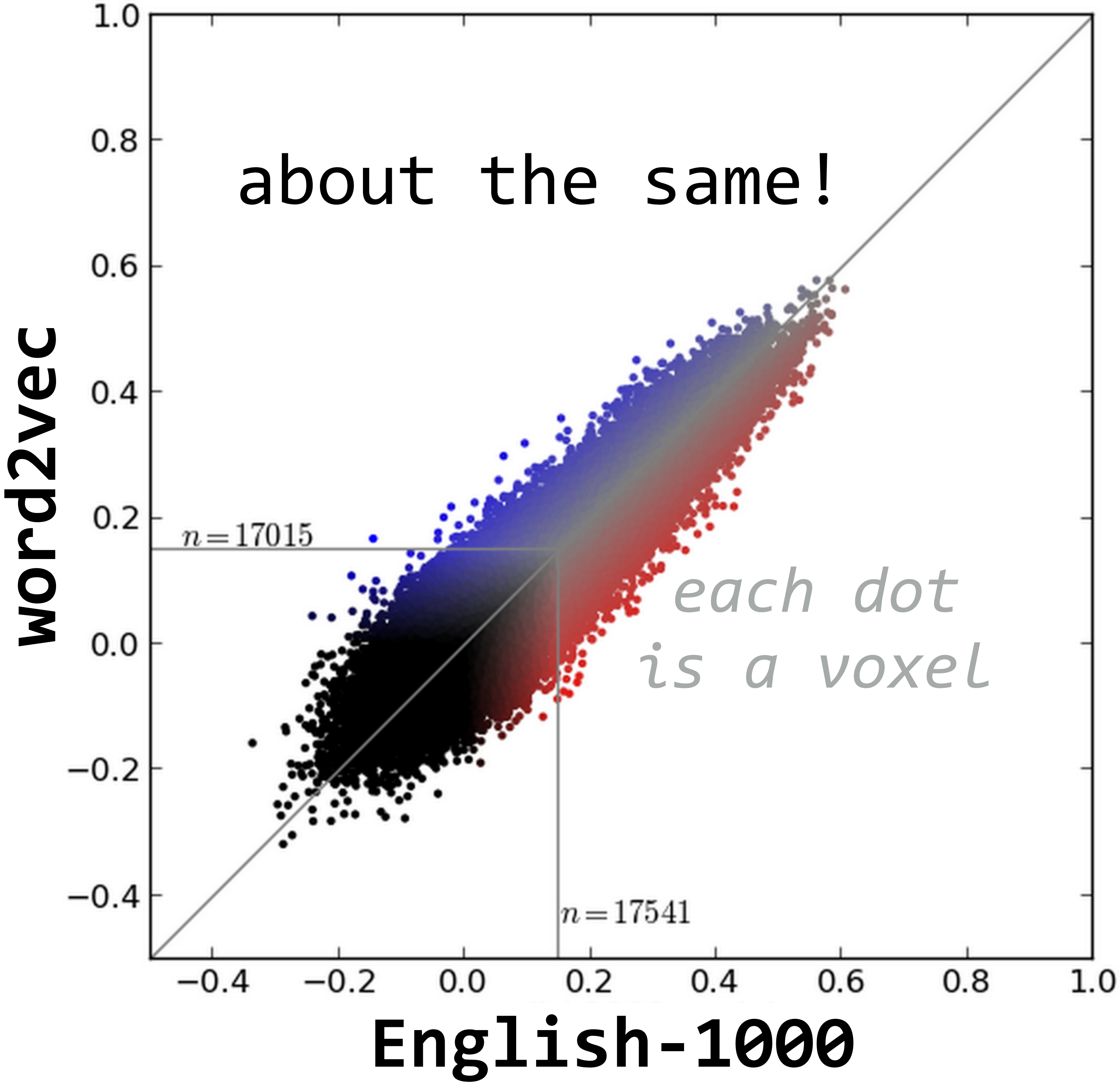


English-1000 better! :)

*each dot is a voxel*

# ENGLISH-1000 VS WORD2VEC

model performance on held-out data



about the same!

*each dot is a voxel*

# NEXT TIME

* Model comparison

* Variance partitioning