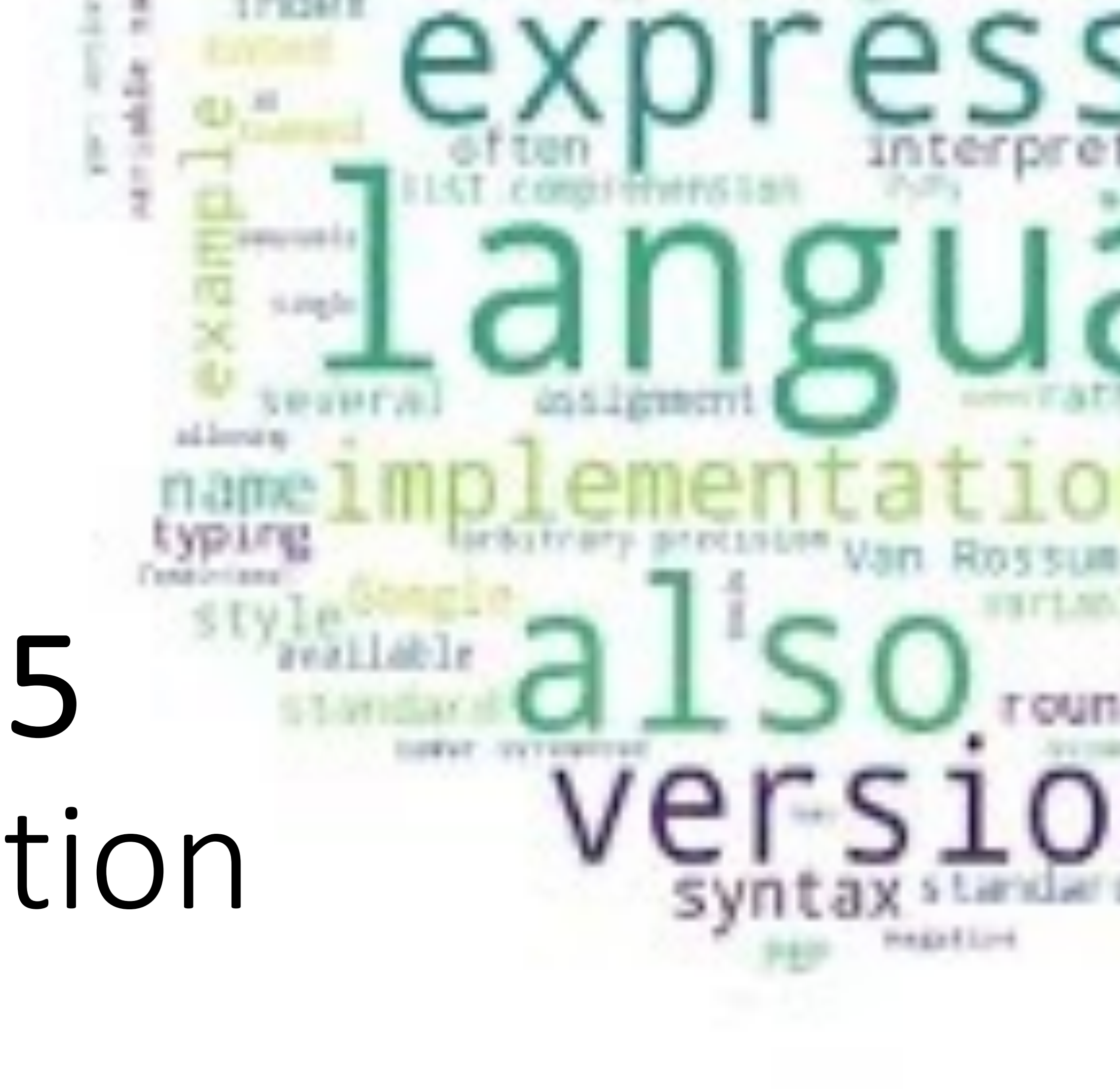
Topic 5
Visualization



Word2vec Notebook

1, window_size

2, model

Doc2Vec

Project Mile stone

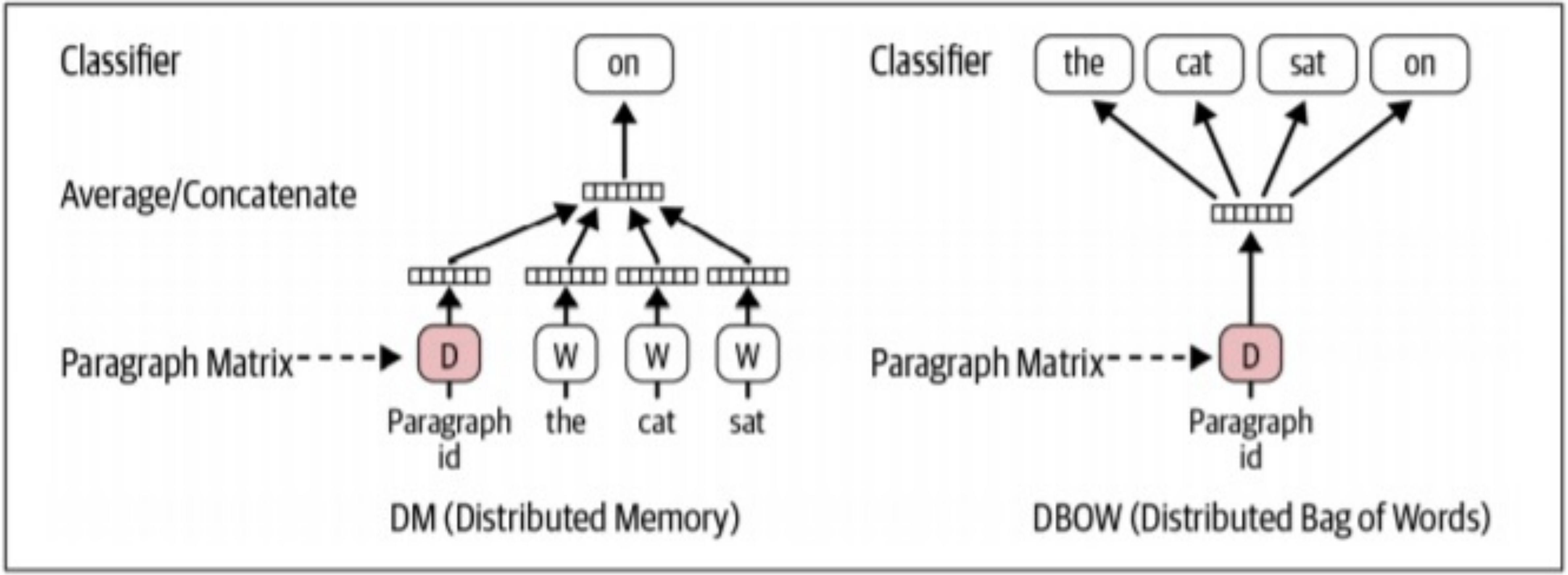


Figure 3-13. Doc2vec architectures: (a) DM and (b) DBOW

Doc 2 Vec

GloVE

GitHub link

Document 1: The farm was home.
Document 2: I liked the farm.

Window size: Let's say it is 3

	the	farm	was	home	I	liked
the	0	2	0	0	0	1
farm	2	0	1	0	0	0
was	0	1	0	1	0	0
home	0	0	1	0	0	0
I	0	0	0	0	0	1
liked	1	0	0	0	1	0

Co-occurrence matrix

GloVE

Document 1: The farm was home.
Document 2: I liked the farm.

Window size: Let's say it is 3

	the	farm	was	home	I	liked
the	0	2	0	0	0	1
farm	2	0	1	0	0	0
was	0	1	0	1	0	0
home	0	0	1	0	0	0
I	0	0	0	0	0	1
liked	1	0	0	0	1	0

Co-occurrence matrix

within our window, "home" appear with "liked" 0 time

# of times the word home occurs in the context of the word liked

Generalization: Each cell is the number of times word j occurs in the context of word I

X_{ij}

GloVE

Document 1: The farm was home.
Document 2: I liked the farm.

Window size: Let's say it is 3

$$\begin{array}{c|cccccc} & \text{the} & \text{farm} & \text{was} & \text{home} & \text{I} & \text{liked} \\ \hline \text{the} & 0 & 2 & 0 & 0 & 0 & 1 \\ \text{farm} & 2 & 0 & 1 & 0 & 0 & 0 \\ \text{was} & 0 & 1 & 0 & 1 & 0 & 0 \\ \text{home} & 0 & 0 & 1 & 0 & 0 & 0 \\ \text{I} & 0 & 0 & 0 & 0 & 0 & 1 \\ \text{liked} & 1 & 0 & 0 & 0 & 1 & 0 \end{array}$$

Co-occurrence matrix

# of times any word appears in the context of word home.

$$X_i = \sum_{j=1}^{N} X_{ij}$$

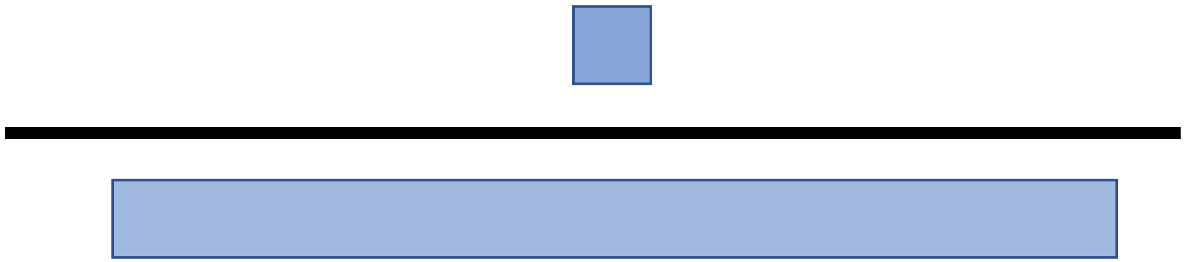
GloVE

Document 1: The farm was home.
Document 2: I liked the farm.

Window size: Let’s say it is 3

	the	farm	was	home	I	liked
the	0	2	0	0	0	1
farm	2	0	1	0	0	0
was	0	1	0	1	0	0
home	0	0	1	0	0	0
I	0	0	0	0	0	1
liked	1	0	0	0	1	0

Co-occurrence matrix



= P(j | i) = The probability word j will appear in the context of word i.

GloVe

- Finding relevant words for the given words among probe words

Probability and Ratio	k = solid	k = gas	k = water	k = fashion
P(k|ice)	1.9 × 10–4	6.6 × 10–5	3.0 × 10–3	1.7 × 10–5
P(k|steam)	2.2 × 10–5	7.8 × 10–4	2.2 × 10–3	1.8 × 10–5
P(k|ice) / P(k|steam)	8.9	8.5 × 10–2	1.36	0.96

Ice and fashion don't frequently appear together

Ice appear with k more than steam

GloVe (alternative of word2vec

$$F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$$

$$F(w_i - w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$$

$$F((w_i - w_j)^T \ \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$$

factorization of the logarithm of co-occurrence matrix

bias

$$\boxed{w_i T \tilde{w}_k + b_i + b_k = log(X_{ik})}$$

- Cost Function = $f\left(X_{ij}\right) * [\sum_{ij=1}^{V} w_i^T \widetilde{w_j} \ + b_i + \ \widetilde{b_j} \ - \ln\left(X_{ij}\right)]^2$

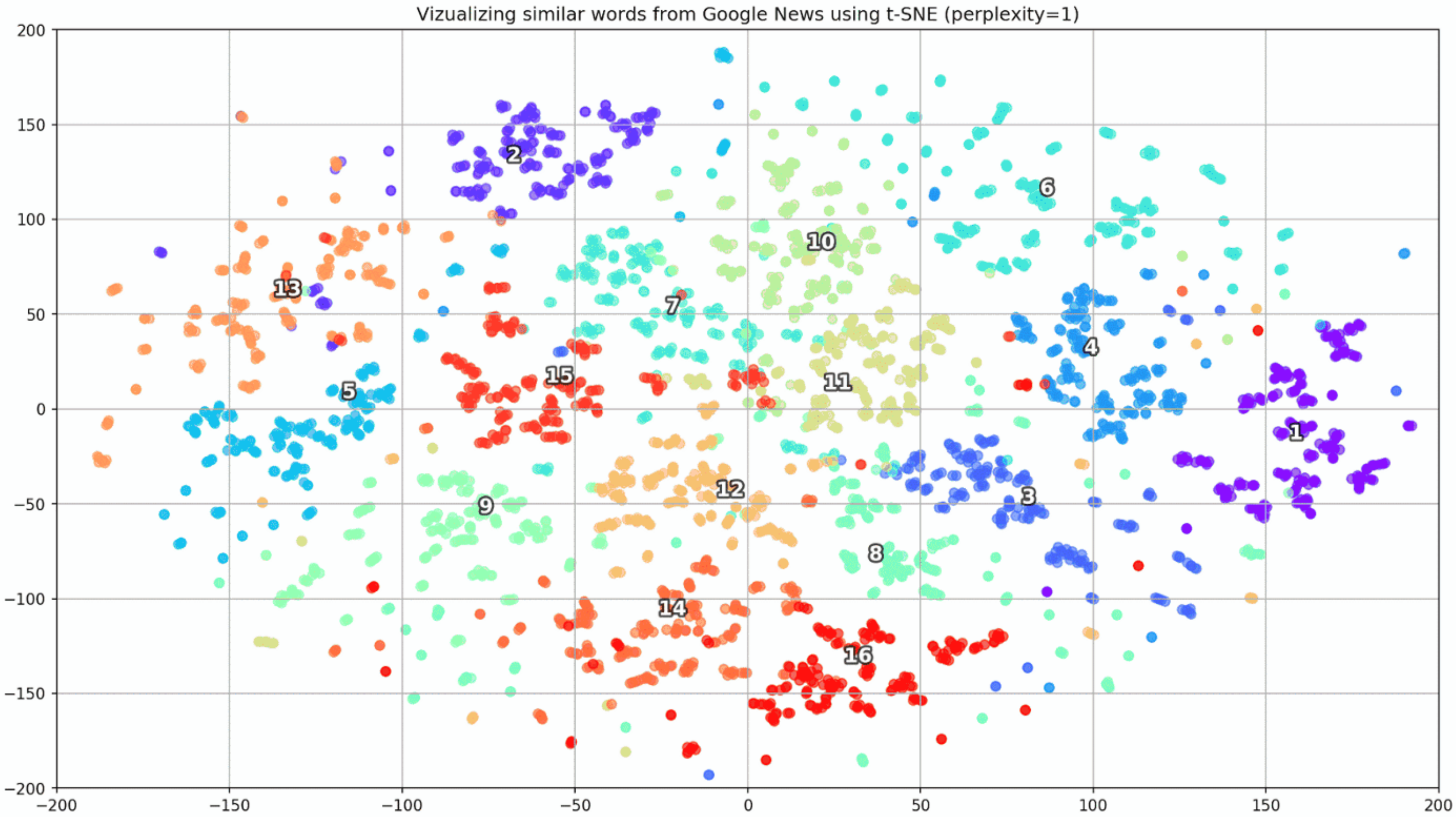
Visually inspecting word vectors

- We still have quite a high-dimensional vector representation for each word…

```
[6] #What is the vector representation for a word?
    w2v_model['beautiful']
```

```
array([-0.01831055,  0.05566406, -0.01153564,  0.07275391,  0.15136719,
       -0.06176758,  0.20605469, -0.15332031, -0.05908203,  0.22851562,
       -0.06445312, -0.22851562, -0.09472656, -0.03344727,  0.24707031,
        0.05541992, -0.00921631,  0.1328125 , -0.15429688,  0.08105469,
       -0.07373047,  0.24316406,  0.12353516, -0.09277344,  0.08203125,
        0.06494141,  0.15722656,  0.11279297, -0.0612793 , -0.296875  ,
       -0.13378906,  0.234375  ,  0.09765625,  0.17773438,  0.06689453,
       -0.27539062,  0.06445312, -0.13867188, -0.08886719,  0.171875  ,
        0.07861328, -0.10058594,  0.23925781,  0.03808594,  0.18652344,
       -0.11279297,  0.22558594,  0.10986328, -0.11865234,  0.02026367,
        0.11376953,  0.09570312,  0.29492188,  0.08251953, -0.05444336,
       -0.0090332 , -0.0625    , -0.17578125, -0.08154297,  0.01062012,
       -0.04736328, -0.08544922, -0.19042969, -0.30273438,  0.07617188,
        0.125     , -0.05932617,  0.03833008, -0.03564453,  0.2421875 ,
        0.36132812,  0.04760742,  0.00631714, -0.03088379, -0.13964844,
        0.22558594, -0.06298828, -0.02636719,  0.1171875 ,  0.33398438,
       -0.07666016, -0.06689453,  0.04150391, -0.15136719, -0.22460938,
        0.03320312, -0.15332031,  0.07128906,  0.16992188,  0.11572266,
       -0.13085938,  0.12451172, -0.20410156,  0.04736328, -0.296875  ,
       -0.17480469,  0.00872803, -0.04638672,  0.10791016, -0.203125  ,
       -0.27539062,  0.2734375 ,  0.02563477, -0.11035156,  0.0625    ,
        0.1953125 ,  0.16015625, -0.13769531, -0.09863281, -0.1953125 ,
       -0.22851562,  0.25390625,  0.00915527, -0.03857422,  0.3984375 ,
       -0.1796875 ,  0.03833008, -0.24804688,  0.03515625,  0.03881836,
```

t-SNE



https://towardsdatascience.com/google-news-and-leo-tolstoy-visualizing-word2vec-word-embeddings-with-t-sne-11558d8bd4d

T-distributed Stochastic Neighbor Embedding

t-SNE

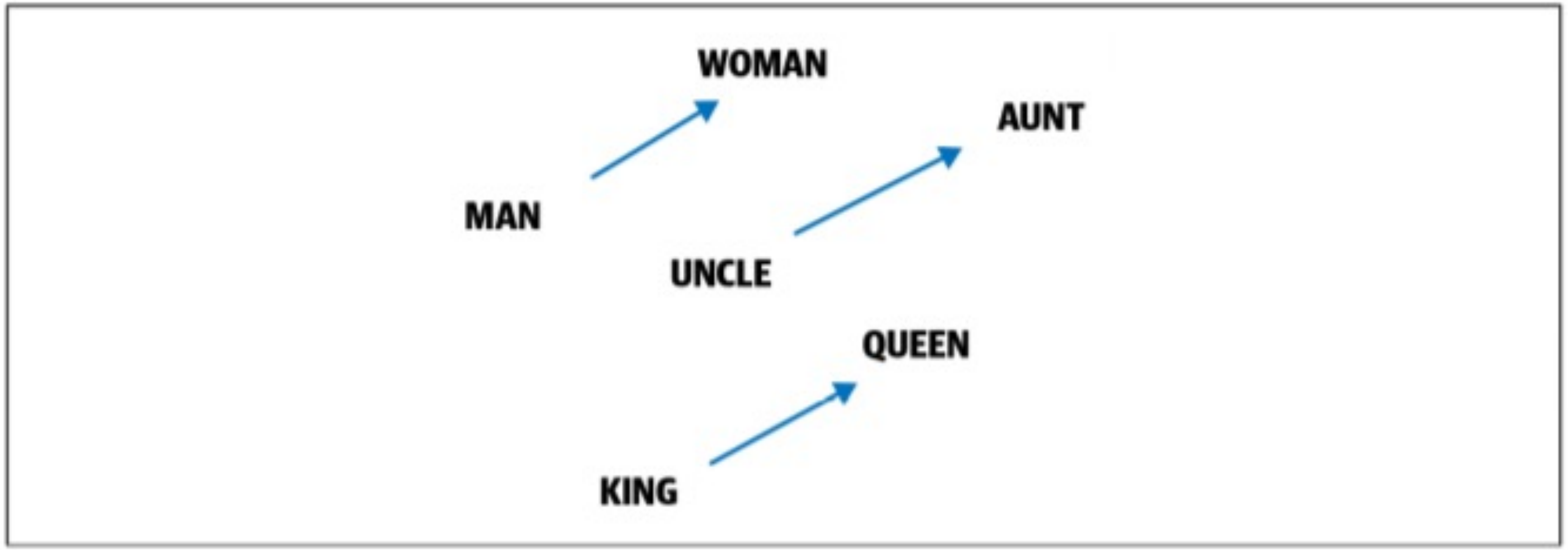
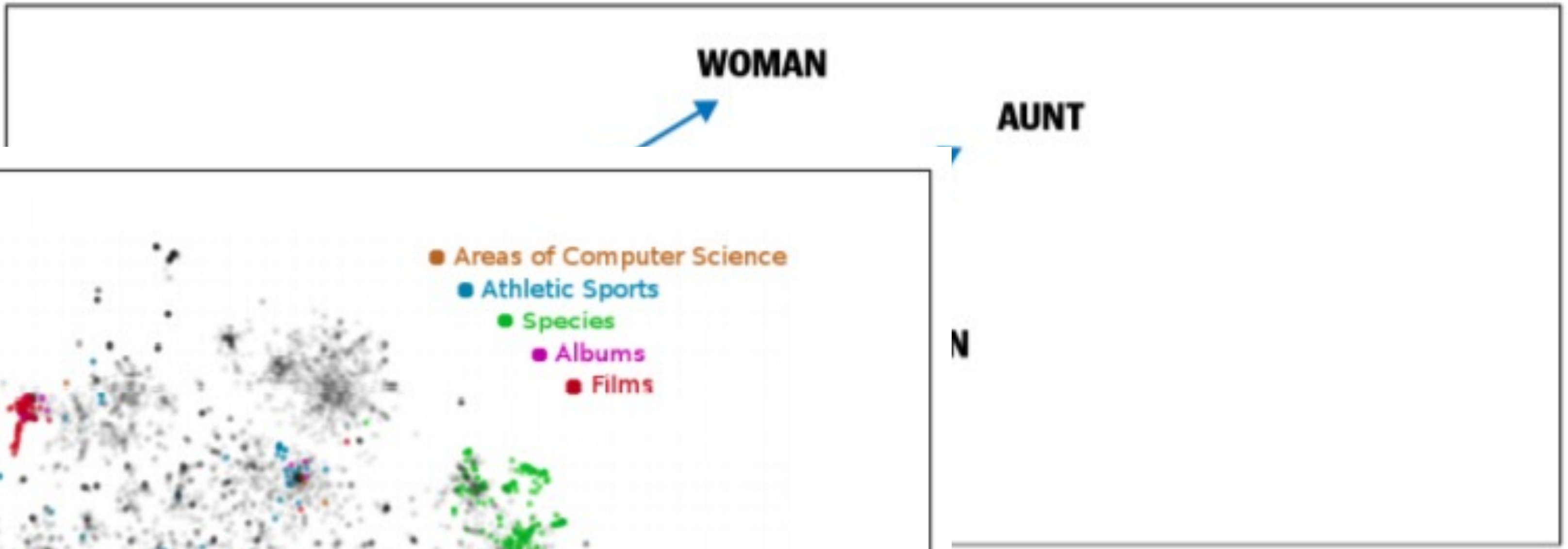


Figure 3-16. t-SNE visualization shows some interesting relationships [7]

t-SNE



ıg relationships [7]

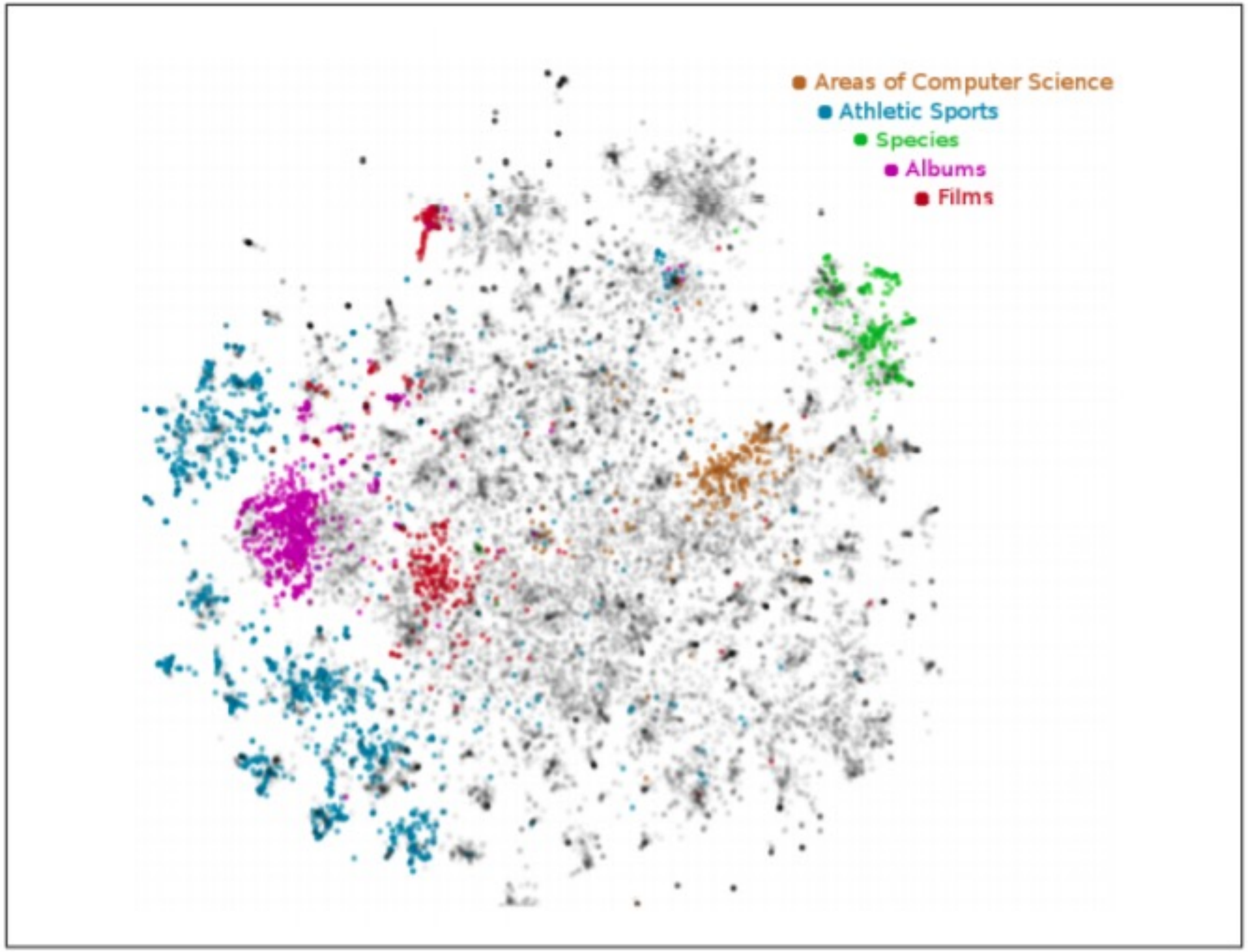
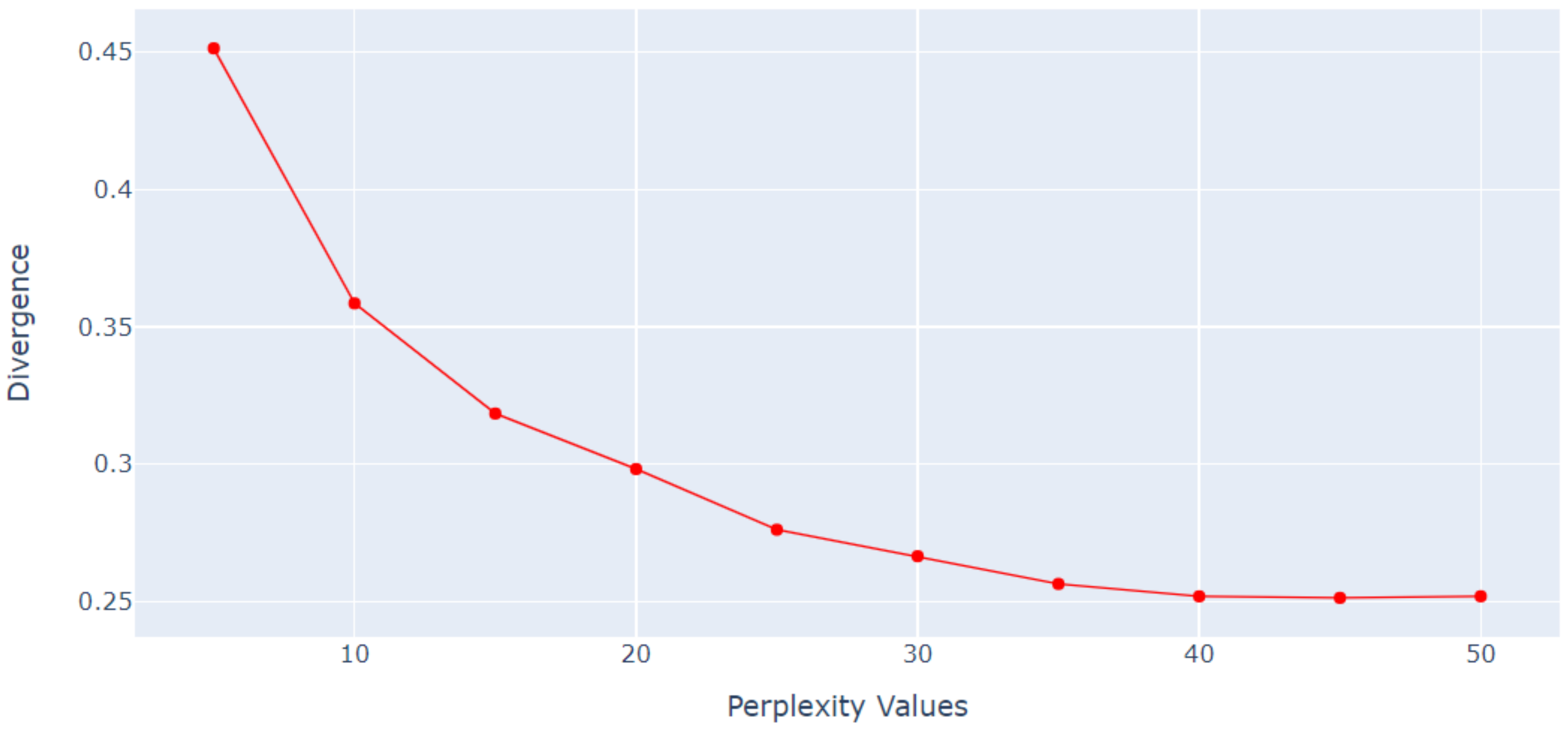


Figure 3-17. Visualization of Wikipedia document vectors [34]

Visualizing Embeddings

- How do we visualize such a high dimension?
- Dimensionality reduction techniques
  - t-SNE
    - *The number of components*
    - *Perplexity value*
    - *The type of initialization*
  - Mapping Gaussian Distribution to a t-distribution

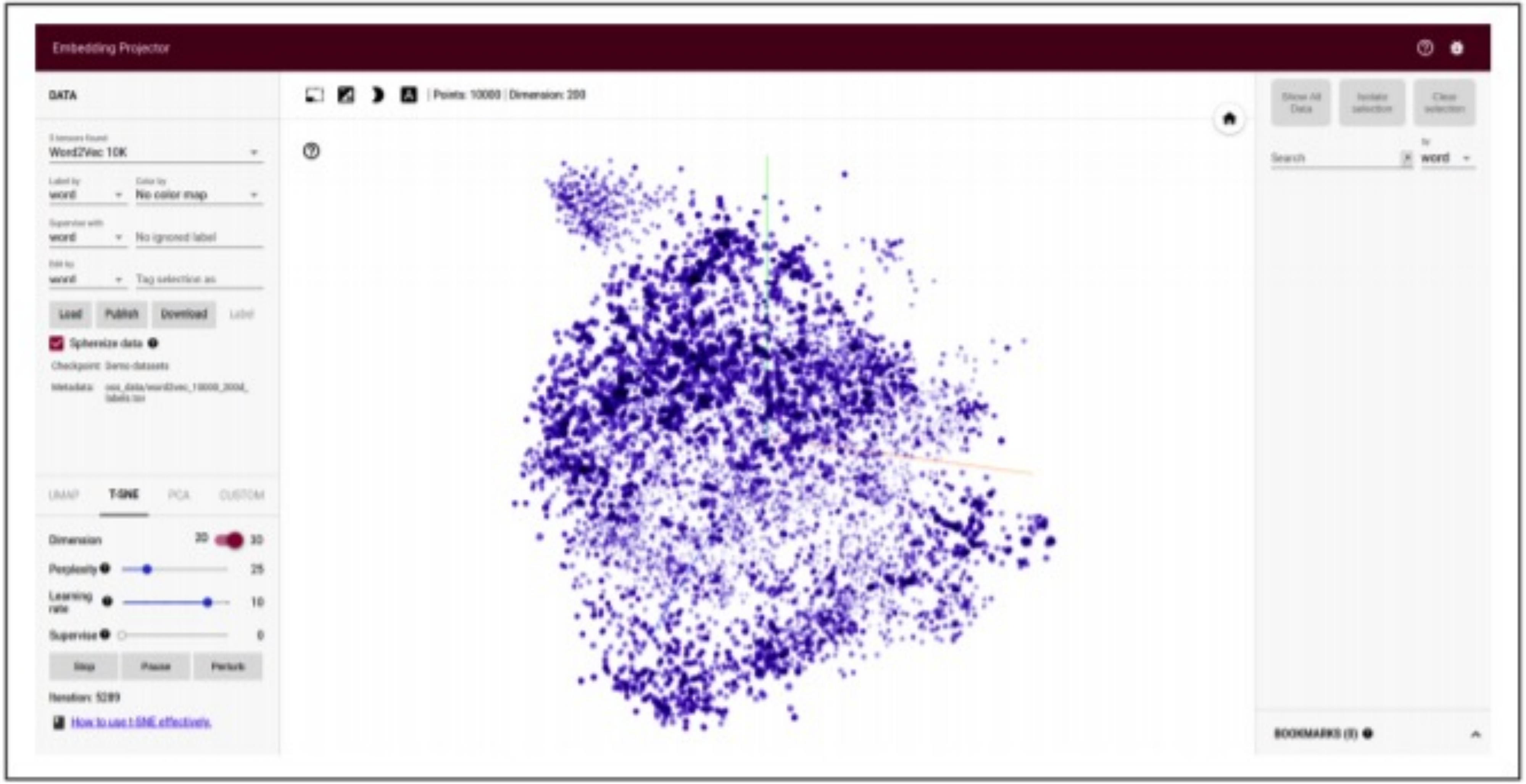


0.45
0.4
0.35
0.3
0.25
Divergence
10
20
30
40
50
Perplexity Values

Steps of t-SNE

1. Compute pairwise similarities between data points
2. Transform similarities into a probability distribution that represents the similarities between datapoints in high-dimensional space
3. Reduce the dimensionality - minimize a cost function measuring difference between the similarities in the high-dimensional space and the low-dimensional space
4. Optimize this mapping between the spaces

The Embedding Projector



https://projector.tensorflow.org

Drawbacks of t-SNE

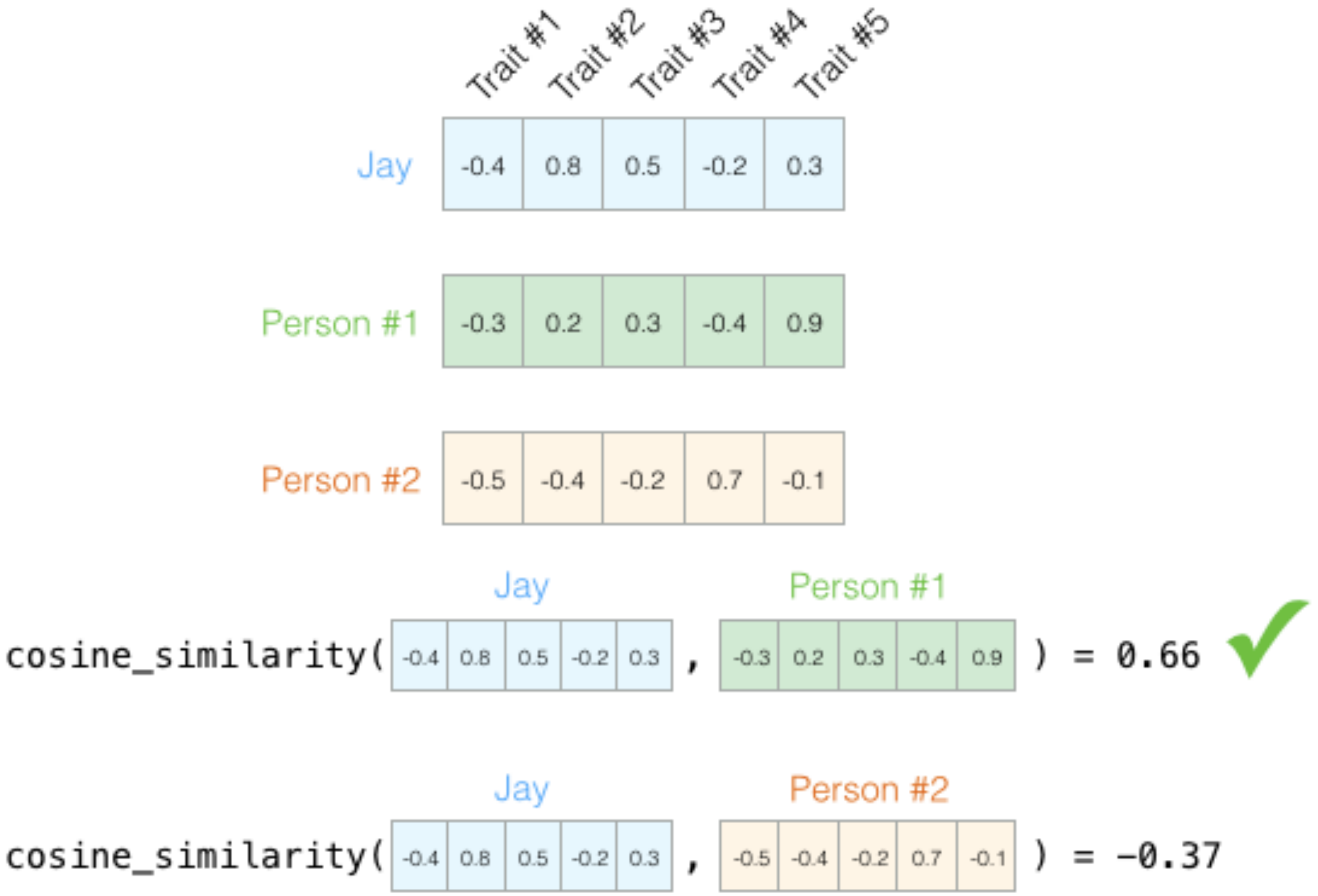
- Perplexity balances the attention t-SNE gives to local and global aspects of the data and can have large effects on the resulting plot
- You cannot see the relative sizes of clusters in a t-SNE plot
- Distances between well-separated clusters in a t-SNE plot may mean nothing
- Clumps of points — especially with small perplexity values — might just be noise

Activity

- Train a Word2Vec model from scratch using Gensim
- Visualize this embedding using t-SNE

Types of Similarity between Embeddings

- Euclidean
- Cosine
- Manhattan
- etc.



Types of Similarity between Embeddings

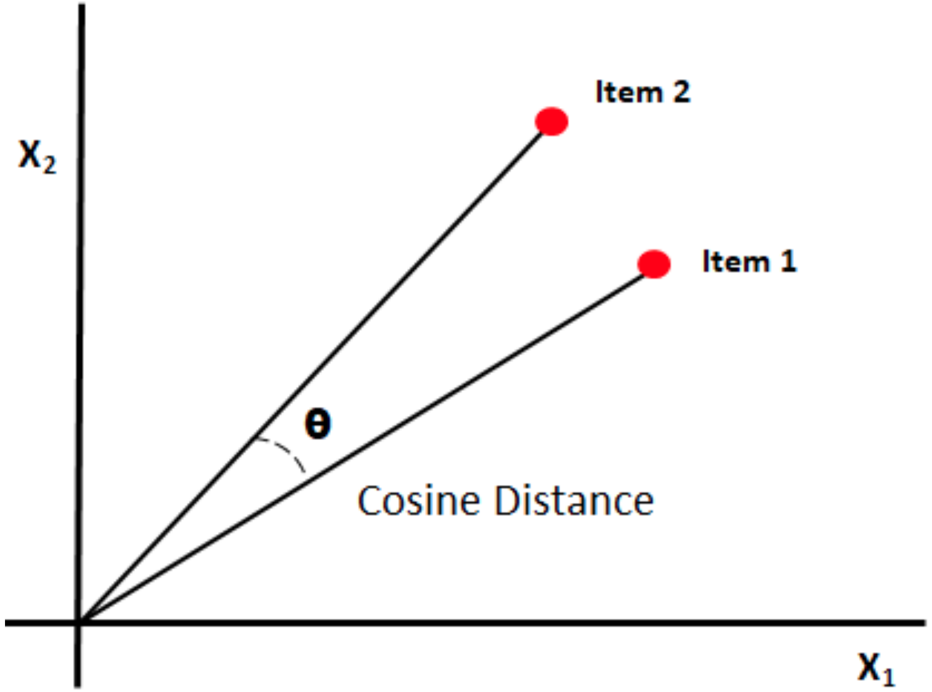
- **Euclidean**
- Cosine
- Manhattan
- etc.

A $= [a_1, \ldots, a_n]$ B $= [b_1, \ldots, b_n]$

euclidean_dist = sqrt($\sum_i (a_i - b_i)^2$)

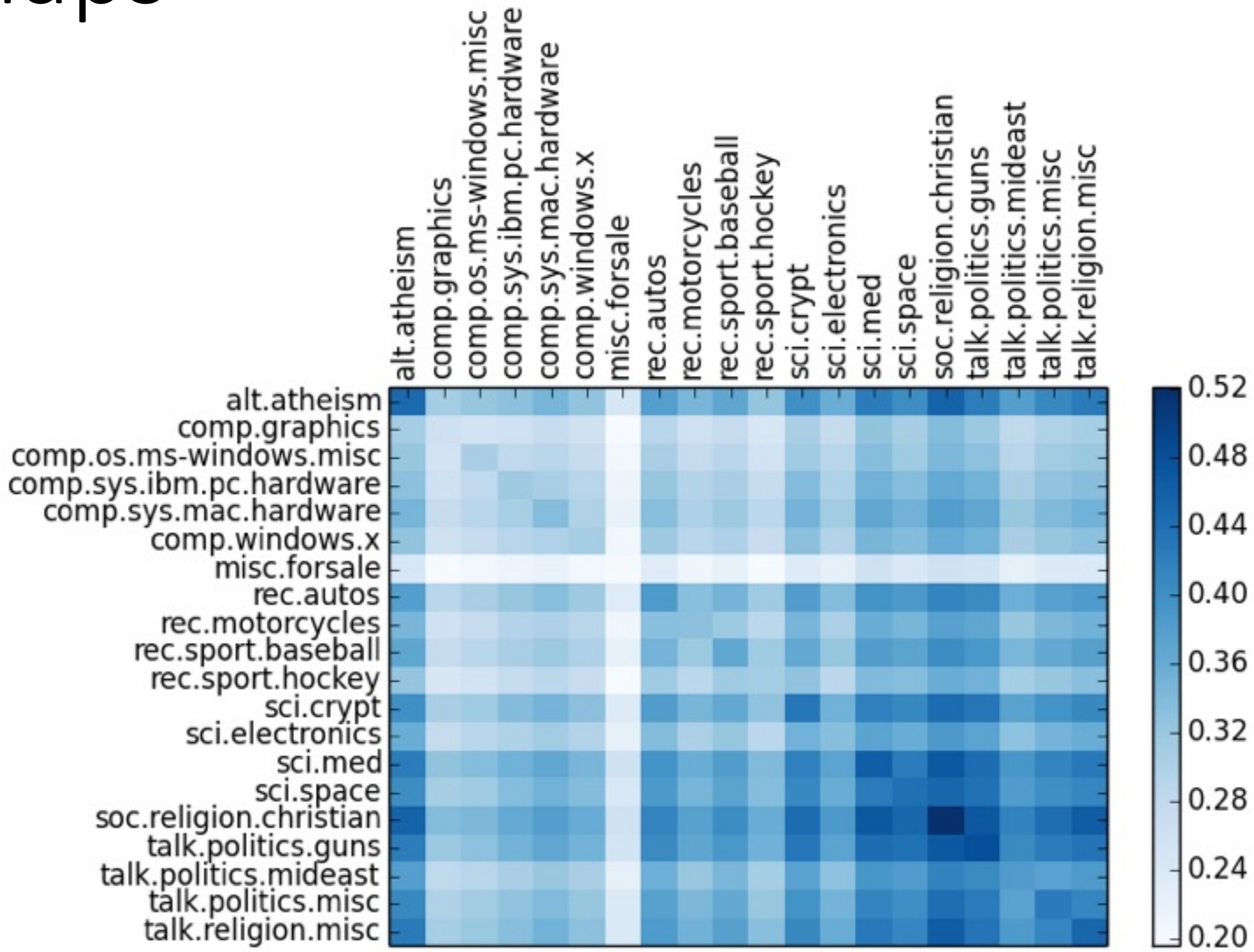
Types of Similarity between Embeddings

- Euclidean
- **Cosine**
- Manhattan
- etc.



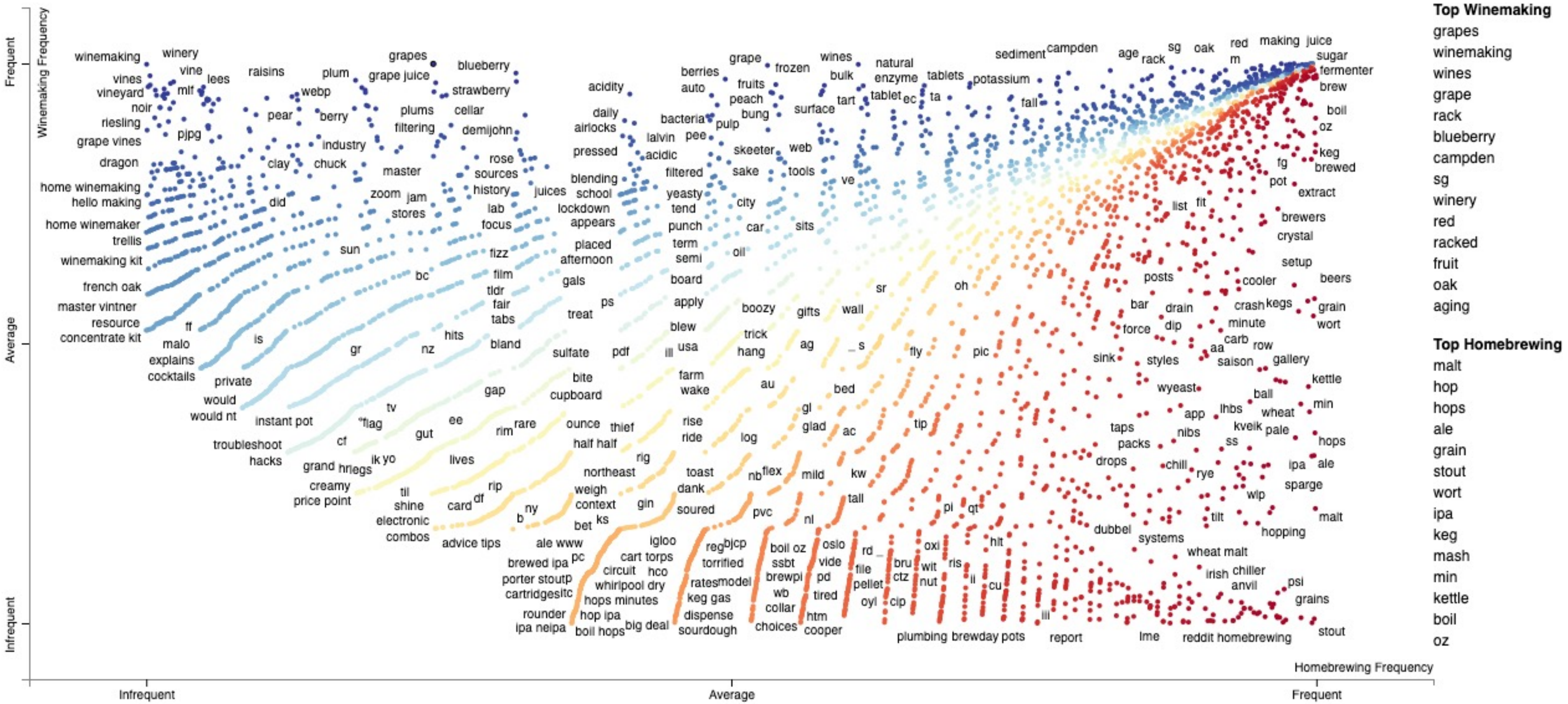
$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Heatmaps



https://www.conniefan.com/2017/03/measures-of-similarity-in-the-20-newsgroups-dataset/

Scattertext (Example in Exercise Notebook)



Next time

- Finding important topics in a text