UNIVERSITÄT LEIPZIG

Toolbox CSS
– Measuring Similarity; Words as Vectors

GWZ H2 1.15, 04/12/2025

Felix Lennert, M.Sc.

OUTLINE

- How to think about “similarity”
- Words as vectors – the Distributional hypothesis
- Properties of these new models
- How we use them in the Social Sciences

BOW HYPOTHESIS

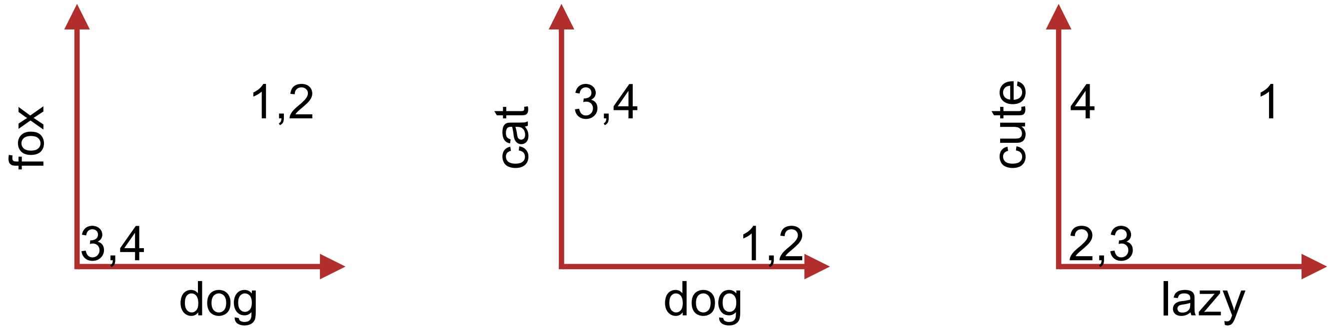
- So far: everything was about the bag-of-words model
- Intuition: document represented by terms it contains
- We can use this for similarity
- Idea: documents are placed in a high-dimensional space based on the words they contain (each word is a dimension)

DOCUMENT SIMILARITY

- Document 1: “The cute fox jumps over the lazy dog”
- Document 2: “The nimble fox jumps over the slow dog”
- Document 3: “Cats are rude animals”
- Document 4: “Cats are cute!”

	fox	dog	cats	animals	cute	lazy	nimble	slow	rude
D 1	1	1	0	0	1	1	0	0	0
D 2	1	1	0	0	0	0	1	1	0
D 3	0	0	1	1	0	0	0	0	1
D 4	0	0	1	0	1	0	0	0	0

DOCUMENT SIMILARITY



"SIMILARITY"

- So how can we think about similarity? ⇒ measure of "distance" in this space
- Two common measures:
  - Euclidean Distance (how distant are these points in "absolute terms")

$$d(\mathbf{u}, \mathbf{v}) = \sqrt{\sum_{i=1}^{n} (u_i - v_i)^2}$$

  - Cosine Similarity (how does their angle from origin differ)

$$\text{cosine\_similarity}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{||\mathbf{u}|| \, ||\mathbf{v}||}$$

EUCLIDEAN DISTANCE

- $$d(\mathbf{u},\mathbf{v}) = \sqrt{\sum_{i=1}^{n}(u_i - v_i)^2}\,.$$
- Idea: If two values are "the same", they do not add to the distance
  $\Longrightarrow$ lower values indicate "closer" points
- $\mathbf{D}_1 = (1,1,0,0,1,1,0,0), \quad \mathbf{D}_2 = (1,1,0,0,0,0,1,1)$
- $d(\mathbf{D}_1,\mathbf{D}_2) = \sqrt{(1-1)^2 + (1-1)^2 + (0-0)^2 + (0-0)^2 + (1-0)^2 + (1-0)^2 + (0-1)^2 + (0-1)^2}$
  $= \sqrt{0+0+0+0+1+1+1+1} = \sqrt{4} = 2.$

EUCLIDEAN DISTANCE

$$
\begin{array}{c}
\begin{array}{ccccc}
 & D_1 & D_2 & D_3 & D_4
\end{array} \\
\begin{array}{c}
D_1 \\ D_2 \\ D_3 \\ D_4
\end{array}
\begin{bmatrix}
0 & 2 & 2.449 & 2 \\
2 & 0 & 2.449 & 2.449 \\
2.449 & 2.449 & 0 & 1.414 \\
2 & 2.449 & 1.414 & 0
\end{bmatrix}
\end{array}
$$

EUCLIDEAN DISTANCE VS. COSINE SIMILARITY

- Problem with Euclidean Distance: document length matters
  - Longer documents might contain certain terms multiple times (if we have a long document containing fox 10 times, this might be less similar to other documents just because of its length)
  - No straight-forward way around this (but see Stoltz & Taylor 2024, p. 173 for a potential workaround)
- Workaround: Cosine similarity looks at “angles” from origin

COSINE SIMILARITY

- cosine_similarity$(\mathbf{u}, \mathbf{v}) = \dfrac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \, \|\mathbf{v}\|}$
- Idea: numerator is high if two vectors have high values on same dimensions (inner product or dot product); we divide by magnitude of vectors (the denominator) to standardize
  $\Longrightarrow$ Higher values indicate higher similarity
- Inner Product:
  $$\mathbf{D}_1 \cdot \mathbf{D}_2 = (1 \cdot 1) + (1 \cdot 1) + (0 \cdot 0) + (0 \cdot 0) + (1 \cdot 0) + (1 \cdot 0) + (0 \cdot 1) + (0 \cdot 1) = 2.$$
- Magnitudes:
  $$\|\mathbf{D}_1\| = \sqrt{1^2 + 1^2 + 0^2 + 0^2 + 1^2 + 1^2 + 0^2 + 0^2} = \sqrt{4} = 2, \|\mathbf{D}_2\| = \sqrt{1^2 + 1^2 + 0^2 + 0^2 + 0^2 + 0^2 + 1^2 + 1^2} = \sqrt{4} = 2.$$
- Cosine Similarity: cosine_similarity$(\mathbf{D}_1, \mathbf{D}_2) = \dfrac{\mathbf{D}_1 \cdot \mathbf{D}_2}{\|\mathbf{D}_1\| \, \|\mathbf{D}_2\|} = \dfrac{2}{2 \cdot 2} = 0.5.$

EUCLIDEAN DISTANCE VS. COSINE SIMILARITY

$$
\begin{array}{c@{}c}
 & \begin{array}{cccc} D_1 & D_2 & D_3 & D_4 \end{array} \\
\begin{array}{c} D_1 \\ D_2 \\ D_3 \\ D_4 \end{array} &
\begin{bmatrix}
1 & 0.5 & 0 & 0.354 \\
0.5 & 1 & 0 & 0 \\
0 & 0 & 1 & 0.5 \\
0.354 & 0 & 0.5 & 1
\end{bmatrix}
\end{array}
$$

THE PROBLEM WITH BOW

- All words are treated the same
  - “dog” and “cat” are as similar as “dog” and “house”
  - “dogs” and “dog” are as similar as “dog” and “house”
    $\Longrightarrow$ we can mitigate the latter by using lemmas/wordstems
- This works fairly well for most tasks
- However, wouldn’t it be great if we could harness more information on the “sense” of words?

DISTRIBUTIONAL HYPOTHESIS

- Was formulated in the 1950s by Firth, can also be traced back to Wittgenstein
- "Words that occur in *similar contexts* tend to have *similar meanings*." (Jurafsky and Martin, forthcoming)
- Word embeddings capture words' contexts instead of the word itself

DISTRIBUTIONAL HYPOTHESIS

Example:

- Ongchoi is delicious sauteed with garlic.
- Ongchoi is superb over rice.
- …ongchoi leaves with salty sauces…

- …spinach sauteed with garlic over rice...
- …chard stems and leaves are delicious...
- …collard greens and other salty leafy greens

⟹ **What do you think does Ongchoi look like?**

DISTRIBUTIONAL HYPOTHESIS

- “Words that occur in *similar contexts* tend to have *similar meanings*.” (Jurafsky and Martin, forthcoming)
- Word embeddings capture words’ contexts instead of the word itself
- Words become **dots in a multidimensional space** (position determined by meaning)

HOW ARE THEY TRAINED

- We want terms which appear in the same contexts to have roughly the same position
- Context is determined by the words that surround a word

is traditionally followed by	**cherry**	pie, a traditional dessert
often mixed, such as	**strawberry**	rhubarb pie. Apple pie
computer peripherals and personal	**digital**	assistants. These devices usually
a computer. This includes	**information**	available on the internet

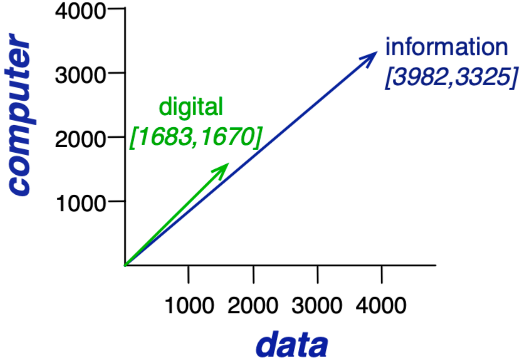
HOW ARE THEY TRAINED

is traditionally followed by	**cherry**	pie, a traditional dessert
often mixed, such as	**strawberry**	rhubarb pie. Apple pie
computer peripherals and personal	**digital**	assistants. These devices usually
a computer. This includes	**information**	available on the internet

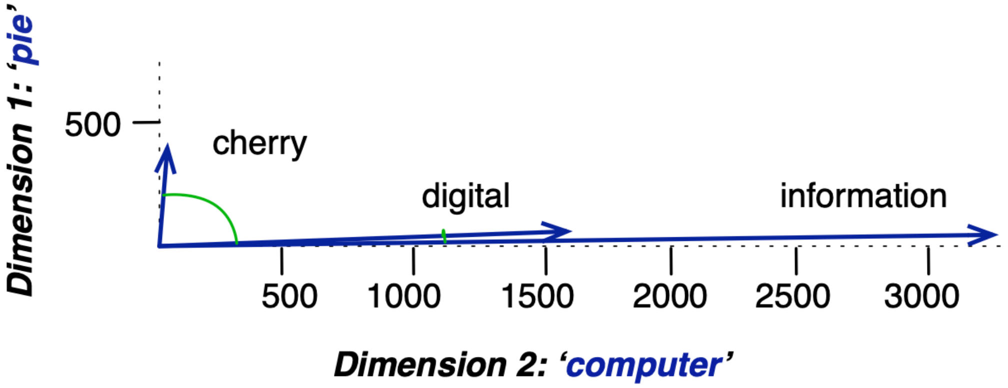
	aardvark	...	computer	data	result	pie	sugar	...
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...

HOW ARE THEY TRAINED

	aardvark	...	computer	data	result	pie	sugar	...
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...



MEASURING SIMILARITY



- Similarity can be assessed by using cosine similarity

MEASURING SIMILARITY

$$|cherry| = \sqrt{2^2 + 442^2},\ |digital| = \sqrt{1670^2 + 5^2},\ |information| = \sqrt{3325^2 + 5^2}$$

Now we can properly compare the values:

$$cosine(cherry, digital) = \frac{2 \times 1670 + 442 \times 5}{\sqrt{2^2 + 442^2} \times \sqrt{1670^2 + 5^2}} = \frac{5590}{\sqrt{195368}\sqrt{2788925}} = 0.007572978$$

$$cosine(information, digital) = \frac{3325 \times 1670 + 5 \times 5}{\sqrt{3325^2 + 5^2} \times \sqrt{1670^2 + 5^2}} = \frac{5552775}{\sqrt{11055625}\sqrt{2788925}} = 0.9999955$$

Cosine similarity is

- 0 if two vectors are in 90° angle (orthogonal)
- 1 if they're perfectly aligned
- -1 if they show in perfectly opposite direction

HOW ARE THEY TRAINED

	aardvark	...	computer	data	result	pie	sugar	...
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...

- Problem with this word-word-matrix: it is quite sparse (i.e., there are many zeroes)
- Solution: reduce its dimensionality (typically to 50-300 dimensions)
- Dimensions have no clear interpretation – but: relationships between words are retained

HOW ARE THEY TRAINED

- Newer applications have different strategies to learn the weights
- But the intuitions still remain the same
- Also, pre-trained embeddings exist that were trained on huge corpora of text ("transfer learning" – using a model that has been trained on a different data source)

- Social scientists have been using these new things in various ways thus far:
  - For better supervised ML classifiers (Bonikowski et al. 2023)
  - To analyze how the meanings of words have shifted (Garg et al. 2018, various things by Laura Nelson and Alina Arseniev-Koehler)
  - For political scaling (Rheault and Cochrane 2018)

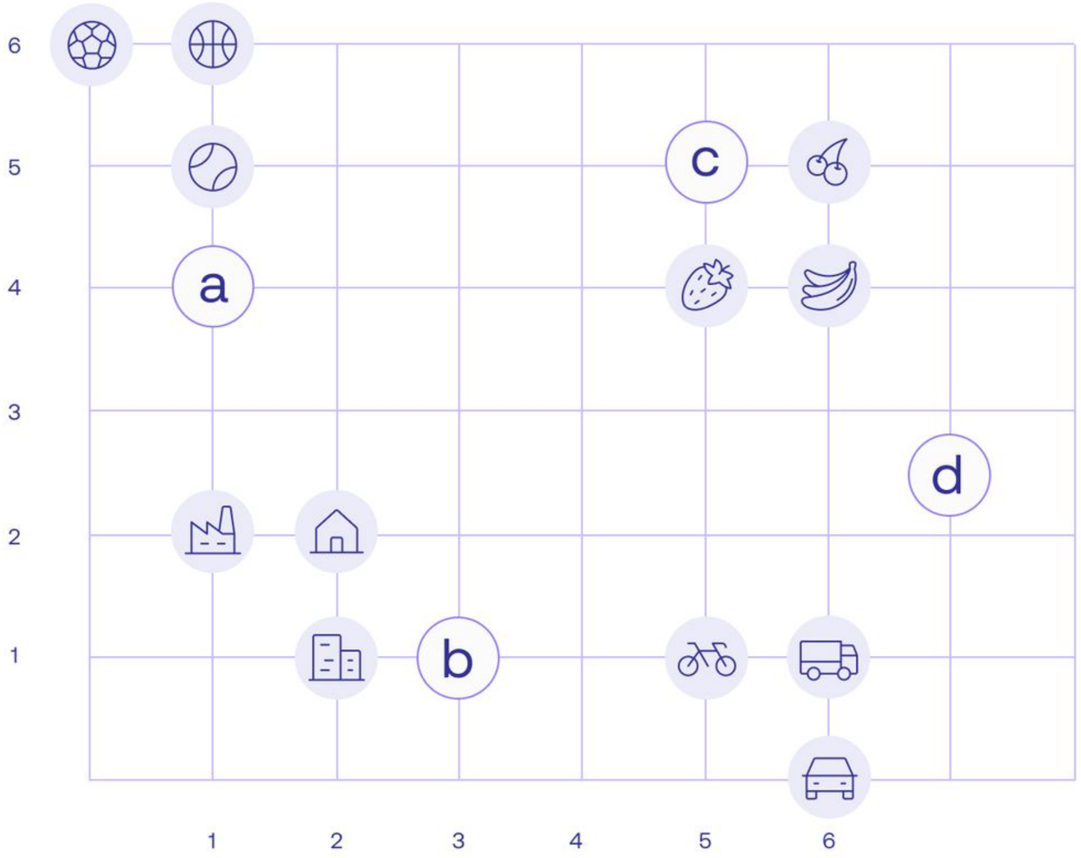
ADVANTAGES OF WORD EMBEDDINGS

Why are they useful for social scientists? (Grimmer et al. 2022)

- They encode similarity,
- They allow for "automatic generalization,"
- They provide a measure of meaning.

Embeddings Quiz 1:

Where would you put the word “apple”?



ADVANTAGES OF WORD EMBEDDINGS

Why are they useful for social scientists? (Grimmer et al. 2022)

- They allow for automatic generalization
  - Big problem for supervised classifiers: it can only learn from the words it has seen before
  - By including (pre-trained) embeddings in the process, the classifier also gets information on words it hasn’t seen before
  - This can also backfire: the social world is unfair and biased; if word embeddings are used for tasks they may reinforce these inequalities

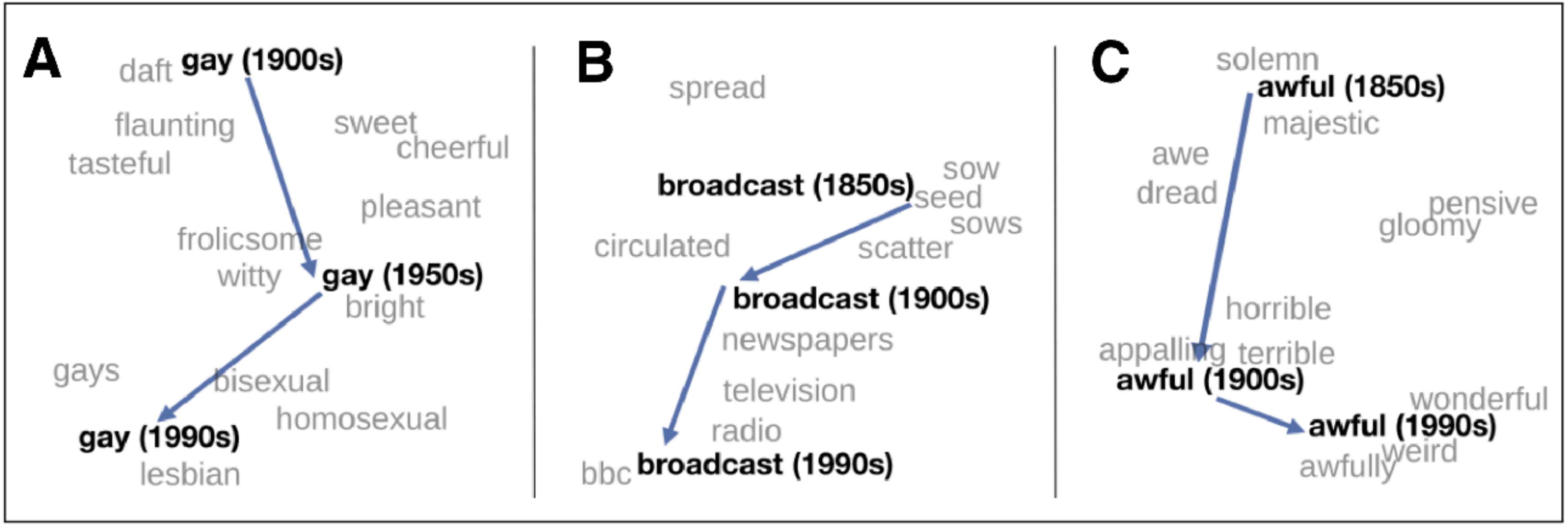
    ⇒ That’s why Computer Scientists need good sociologists 😏

ADVANTAGES OF WORD EMBEDDINGS

Why are they useful for social scientists? (Grimmer et al. 2022)

- they provide a measure of meaning.
  - We can compare the relationships of words over time and authors/ speakers
  - Latent higher-order relationships are retained, too, enabling us to answer questions in a new way

WORD MEANING OVER TIME



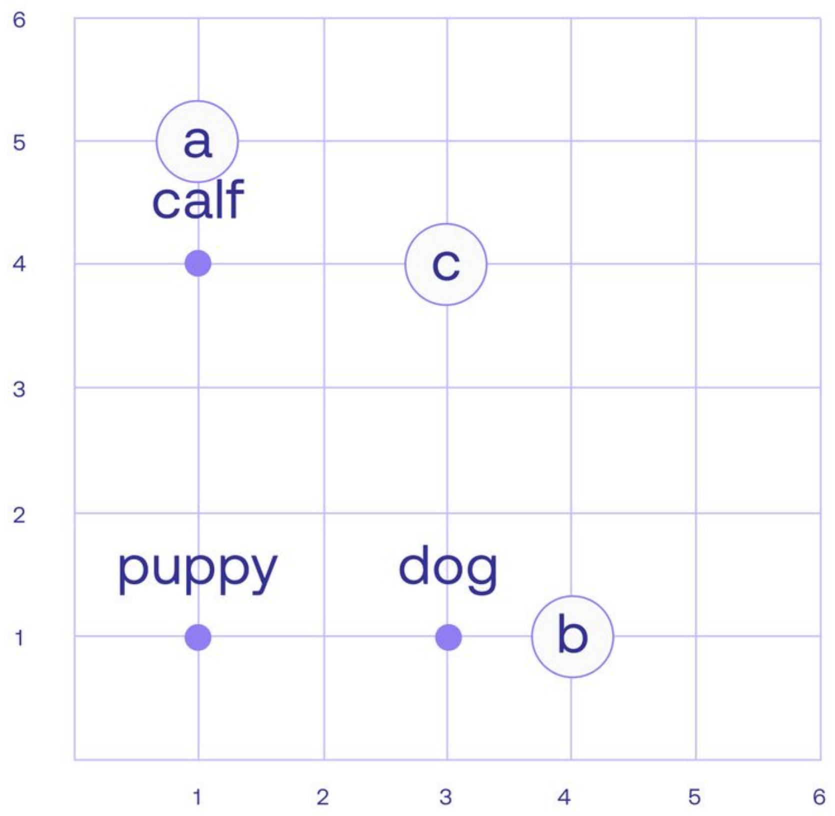
ADVANTAGES OF WORD EMBEDDINGS

Why are they useful for social scientists? (Grimmer et al. 2022)

- they provide a measure of meaning.
  - We can compare the relationships of words over time and authors/ speakers
  - Latent higher-order relationships are retained, too, enabling us to answer questions in a new way

Embeddings Quiz 2:

Where would you put the word "cow"?

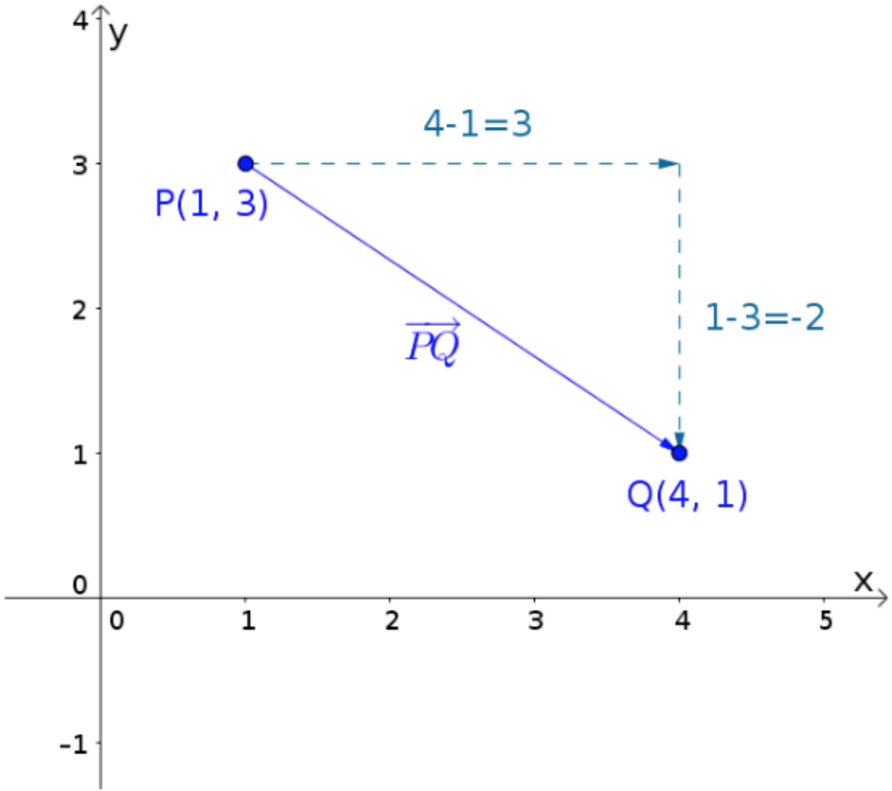


ADVANTAGES OF WORD EMBEDDINGS

Why are they useful for social scientists? (Grimmer et al. 2022)

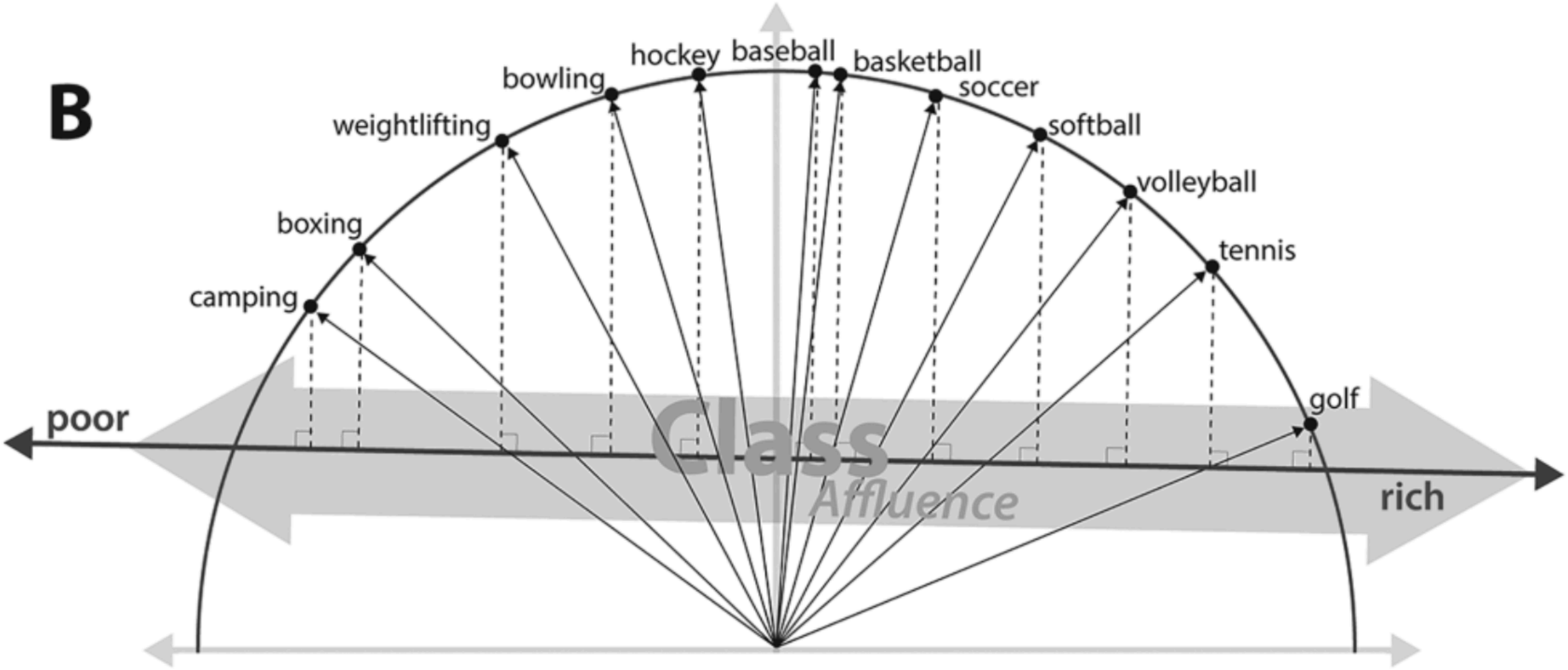
- They encode similarity
  - Two words are very similar if they appear interchangeably (synonyms)
  - Also, higher-order relationships are captured

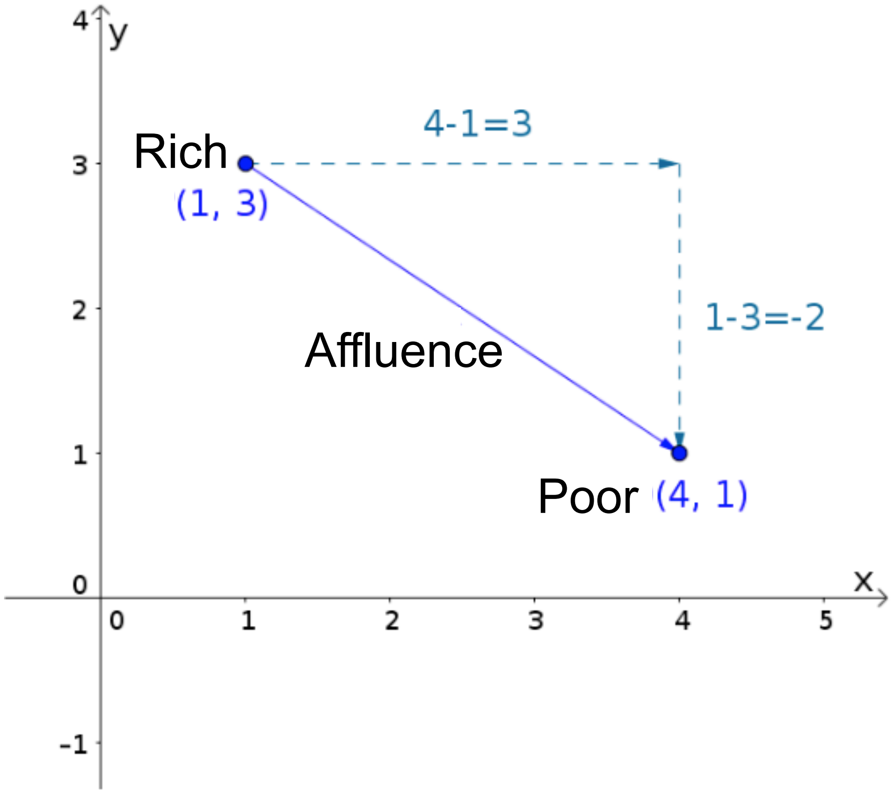
$$\overrightarrow{Paris} - \overrightarrow{France} = ? - \overrightarrow{Italy}$$
$$\overrightarrow{Paris} - \overrightarrow{France} + \overrightarrow{Italy} = ?$$
$$\overrightarrow{Paris} - \overrightarrow{France} + \overrightarrow{Italy} \approx \overrightarrow{Rome}$$



$$\overrightarrow{PQ} = \begin{pmatrix} 4-1 \\ 1-3 \end{pmatrix} = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$$

ADVANTAGES OF WORD EMBEDDINGS





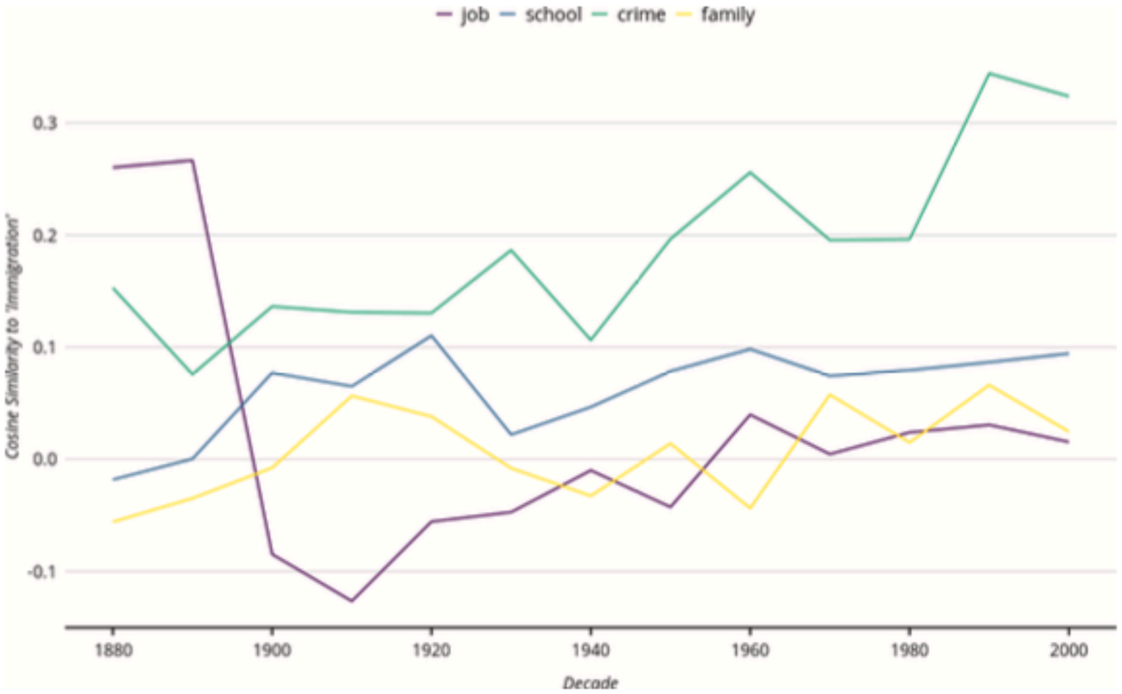
$$\overrightarrow{PQ} = \begin{pmatrix} 4-1 \\ 1-3 \end{pmatrix} = \begin{pmatrix} 3 \\ -2 \end{pmatrix}$$

VARIABLE VS. FIXED EMBEDDING SPACES (STOLTZ & TAYLOR 2021)

- Variable Embedding Space: train multiple models on sub-corpora and compare them
  - compare word similarities over time
  - potential challenge: embedding spaces need to be aligned (if you want to compare how word meanings change in relation to all other words)
  - e.g., comparisons of word meaning over time, per author

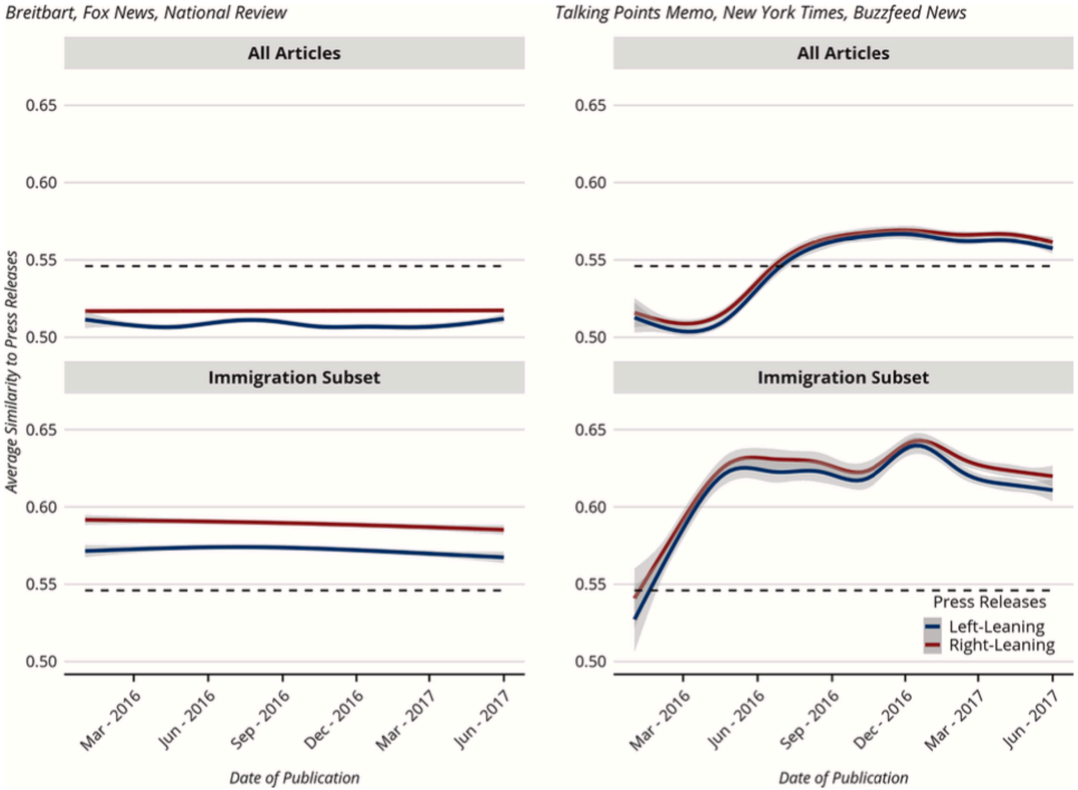
- Fixed Embedding Space: use one embedding space for the entire corpus
  - embed documents in this space (usually using pre-trained models)
    i.e., take all words within one document – extract their vectors – use centroid of the document (average of all vectors)
  - e.g., comparison of document similarities, concept engagement

VARIABLE SPACES – APPLICATIONS (STOLTZ & TAYLOR 2021)

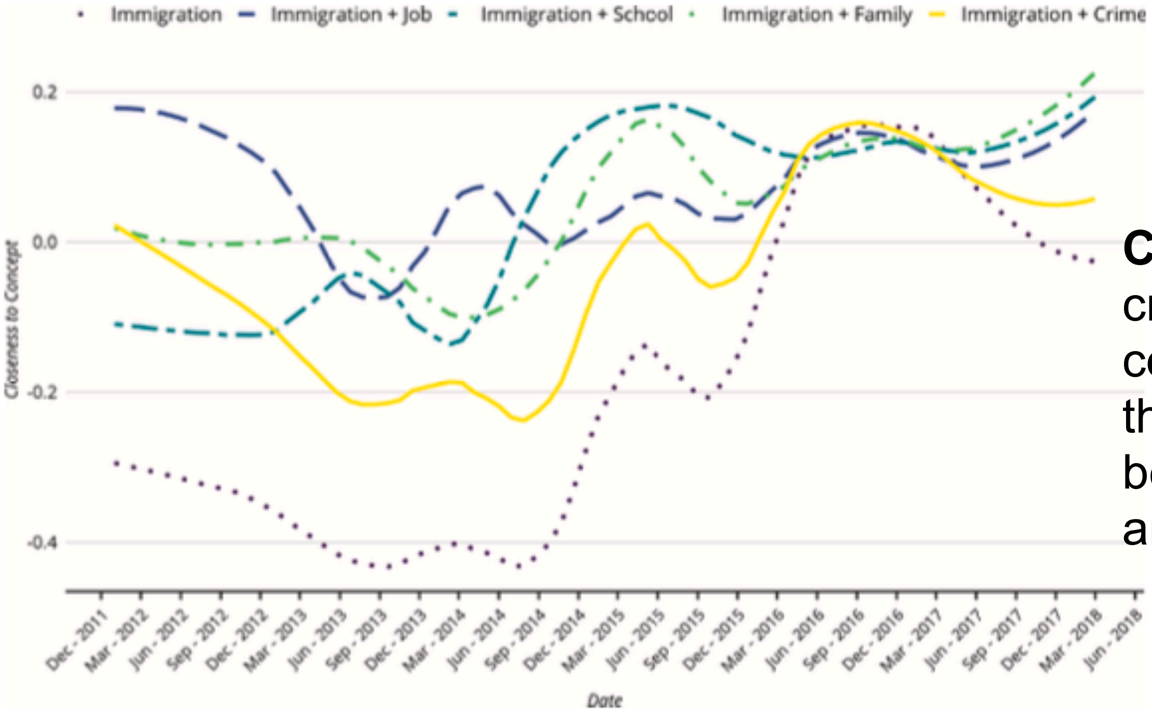


Cosine Similarity of 'Immigration' and Key Terms by Decade, 1880 to 2000.

FIXED SPACE – APPLICATIONS (STOLTZ & TAYLOR 2021)



FIXED SPACE – APPLICATIONS (STOLTZ & TAYLOR 2021)



Concept Mover's Distance (CMD) creates a document that contains a certain concept, then measures the similarity between the "concept" document and the documents in question

Fig. 4. News Articles' Conceptual Engagement Over Time (with CMD).

OUTCOME MEASURES

- You get a measure of similarity/distance
  - Do words bear the same meaning (synonyms or some higher-order relationship)
  - How does a word score on some latent construct (e.g., class, positive-negative, gender)
  - What’s the similarity between certain documents
- These can be connected to document variables
  - author, time, outlet, political leaning of author/outlet, etc.

WHAT'S NEXT

- The latest models (ElMo, BERT) can now also take context into account: vectors of the same word may vary depending on which words they are surrounded by
  - Examples: bank–money ↔ bank–river; cell–prison ↔ cell–phone
  - Makes for more accurate predictions
- This also facilitates language generation – GPT (generative pre-trained transformers)

⟹ Next week

WHAT I WOULD SUGGEST YOU TO READ NEXT IF YOU WANT TO WORK WITH THESE THINGS

- You need to test your hypotheses; this recent paper by Rodriguez et al. (2023) provides you with a method to perform hypothesis tests with embeddings
- These papers deal with the limitations: Arseniev-Kohler (2022), Rodriguez and Spirling (2022)
- Stoltz and Taylor (2021) and Stoltz and Taylor (2024) – chapter 11
- The chapters 7 and 8 in Grimmer et al. (2022) are a thorough introduction; also chapter 6 in Jurafsky and Martin (forthcoming)
- A paper by Bender et al. (2021) on the “dangers of stochastic parrots”

REFERENCES

- Bender, Emily, Timnit Gebru, Angelina McMillan-Major, Shmargaret Shmitchell. 2021. “On the Dangers of Stochastic Parrots: Can Language Models be too Big?,” *ACM Conference on Fairness, Accountability, and Transparency (ACM FAccT*) ’21.
- Garg, Nikhil, Londa Schiebinger, Dan Jurafsky, and James Zou. 2018. “Word Embeddings Quantify 100 Years of Gender and Ethnic Stereotypes.” *Proceedings of the National Academy of Sciences* 115(16):3635–44.
- Grimmer, Justin, Margaret Roberts, and Brandon Stewart. 2022. *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton: Princeton University Press.
- Stoltz, Dustin S. and Marshall A. Taylor. 2021. “Cultural Cartography with Word Embeddings.” *Poetics* 88.

UNIVERSITÄT
LEIPZIG

MERCI

Felix Lennert
Institut für Soziologie

felix.lennert@uni-leipzig.de
www.uni-leipzig.de