



PennState
College of the
Liberal Arts

C-SODA
Center for Social Data Analytics

Day 9 - Natural Language Understanding, Semantic Change, Fairness & Bias, Custom NER

Advanced Text as Data: Natural Language Processing
Essex Summer School in Social Science Data Analysis

Burt L. Monroe (Instructor) & Sam Bestvater (TA)
Pennsylvania State University

August 5, 2021

Today

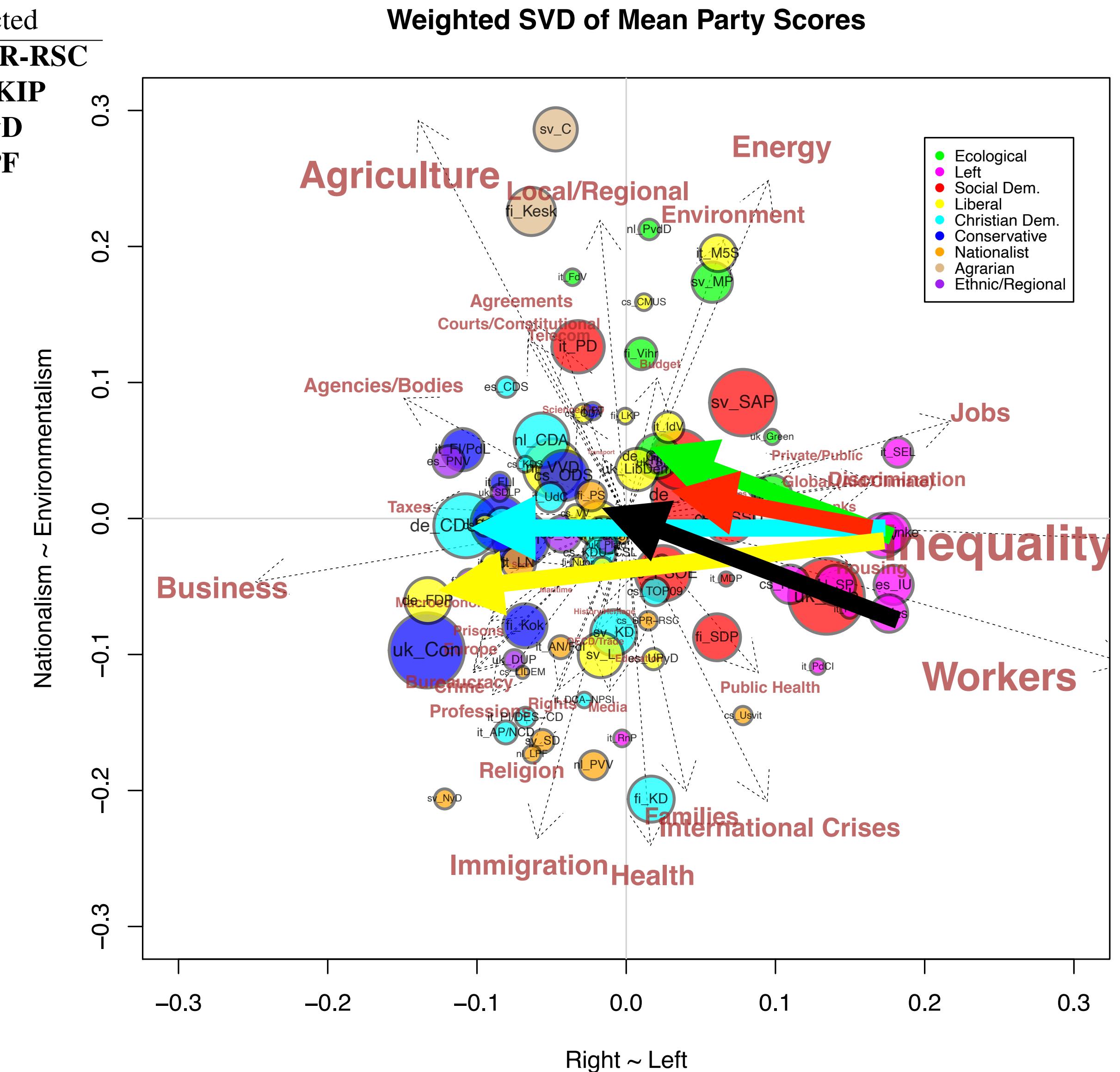
- Lisa's question about the party analogy task.
- Custom NER task / NER annotations (Sukayna, Jūratė, Max)
- Lexical and semantic change (~ Jūratė)
- Bias and fairness in NLP
- Natural language *understanding*?
 - Julia's task (Extent to which company sustainability report follows recommended guidelines.)
 - BERTology
 - Bender & Koller

Lisa's question

fi_Vas (Left Alliance) is to fi_PS (True Finns) as		Cosine	Expected
cs_KSCM (Communist Party)	cs_SPR-RSC (Rally for the Republic)	0.37	cs_SPR-RSC
uk_Plaid (Plaid Cymru)	uk_UKIP (Independence Party)	0.23	uk_UKIP
sv_V (Left)	sv_NyD (New Democracy)	0.22	sv_NyD
nl_GL (Green Left)	nl_LPF (Pim Fortuyn)	0.22	nl_LPF
it_SEL (Left Ecology Freedom)	it_M5S (Five Star)	0.17	it_LN
No nationalist party target:			
es_IU (United Left)	es_CDS (Democratic and Social Centre)	0.19	None
de_Gruene (Green)		0.16	None

?

1. It's random? Somebody's got to be "closest" and there aren't that many to choose from. The cosine similarity is the lowest "match" at .16.
2. Political sciencey answer — it's right? Greens and Nationalists dominate the cultural/post-material dimension and therefore are "closer" in the space than it appears when you focus on economic "left-right." ?
3. It was a bad example because the True Finns are atypical of nationalists.
4. It was a bad example because it captures multiple ideological shifts and we should replace that in the table with the center-right to nationalist analogy we use in the presentation.



LSTM/CNN Notebooks

Custom NER

<https://github.com/tecoholic/ner-annotator>

The screenshot shows the GitHub repository page for 'tecoholic / ner-annotator'. The repository has 175 stars and 43 forks. The 'Code' tab is selected, showing a list of commits from 'tecoholic' and other contributors. A preview of the README.md file is shown below, featuring a screenshot of the 'NER Text Annotator' interface.

Code Issues 7 Pull requests 1 Actions Projects Wiki Security Insights

Go to file Add file Code

About

Named Entity Recognition (NER)
Annotation tool for Spacy. Generates
Traning Data as a JSON which can be
readily used.

ui spacy ner spacy-nlp

Readme MIT License

Contributors 2

tecoholic Arunmozhi Totalus

Languages

Vue 62.3% JavaScript 32.5%
Python 2.8% HTML 2.1%
SCSS 0.3%

ner-annotator-m....zip Show All

Starting the application

1. Install the dependencies and start the Python Backend server

```
python -m venv env
source env/bin/activate
pip install -r requirements.txt
python annotator/server.py
```

2. Open another terminal and start the server for the UI

```
cd ui
yarn install
yarn serve
```

Now go to <http://localhost:8080>

NER Annotator for SpaCy

localhost:8080

Update

NER Text Annotator

Annotate text for SpaCy NER Model training

Select file to start annotating



How to use?

- 1 Prepare and upload the input file**

Put all the text that needs to be annotated into a text file. If the corpus is large, split it into multiple files.
Each line will be presented as an entry for annotating one by one.
- 2 Create Tags and annotate**

You can create any number of custom tags to annotate your text. The text will be presented as tokens for easy tagging.
You can start your selection anywhere on a word and end anywhere to tag a word. No character level accuracy is needed.
- 3 Export your annotations as JSON**

The annotations are exported into a JSON array with the format:

training_data.json

Show All

Demonstrate NER Annotator

[https://github.com/amrrs/custom-ner-with-spacy/blob/main/
pvr custom ner training2.ipynb](https://github.com/amrrs/custom-ner-with-spacy/blob/main/pvr%20custom%20ner%20training2.ipynb)

Colab notebook

Lexical change - how have we changed what words we use?

Science, 2011 (2000+ cites)

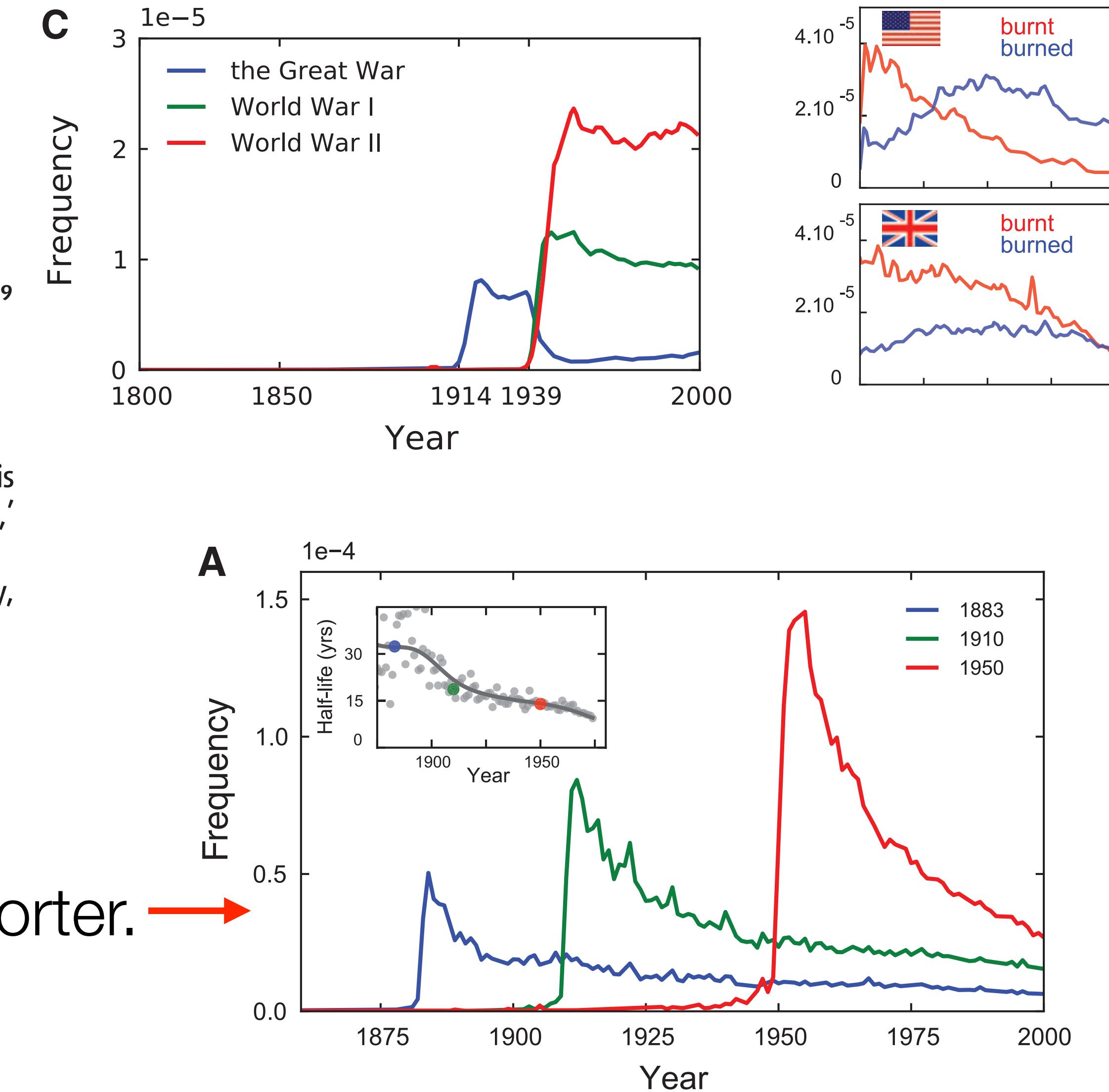
Quantitative Analysis of Culture Using Millions of Digitized Books

Jean-Baptiste Michel,^{1,2,3,4,5*}† Yuan Kui Shen,^{2,6,7} Aviva Presser Aiden,^{2,6,8} Adrian Veres,^{2,6,9} Matthew K. Gray,¹⁰ The Google Books Team,¹⁰ Joseph P. Pickett,¹¹ Dale Hoiberg,¹² Dan Clancy,¹⁰ Peter Norvig,¹⁰ Jon Orwant,¹⁰ Steven Pinker,⁵ Martin A. Nowak,^{1,13,14} Erez Lieberman Aiden^{1,2,6,14,15,16,17,*†}

We constructed a corpus of digitized texts containing about 4% of all books ever printed. Analysis of this corpus enables us to investigate cultural trends quantitatively. We survey the vast terrain of ‘culturomics,’ focusing on linguistic and cultural phenomena that were reflected in the English language between 1800 and 2000. We show how this approach can provide insights about fields as diverse as lexicography, the evolution of grammar, collective memory, the adoption of technology, the pursuit of fame, censorship, and historical epidemiology. Culturomics extends the boundaries of rigorous quantitative inquiry to a wide array of new phenomena spanning the social sciences and the humanities.

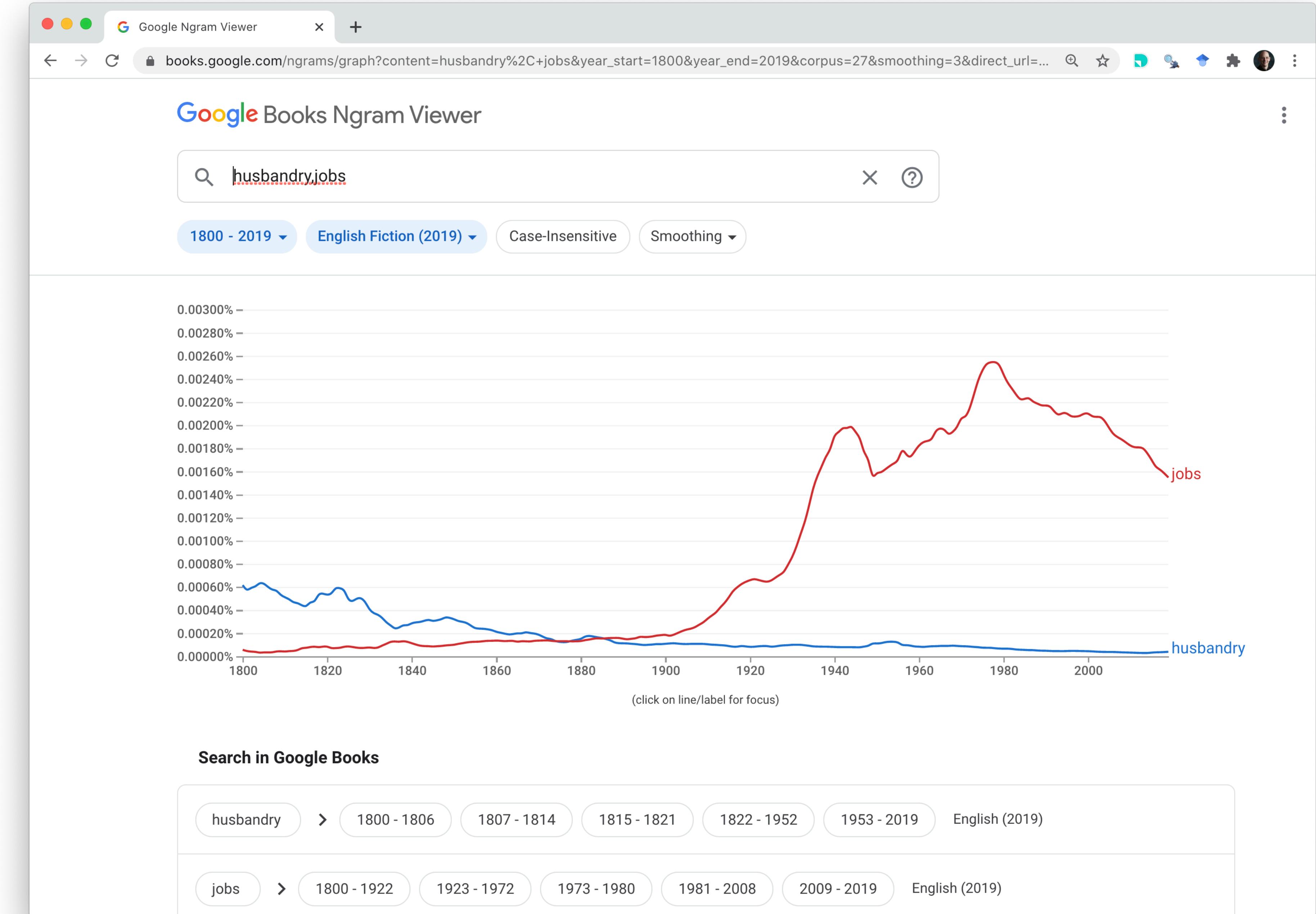
They argue that cultural memory is getting shorter. →

Try it! <https://books.google.com/ngrams>

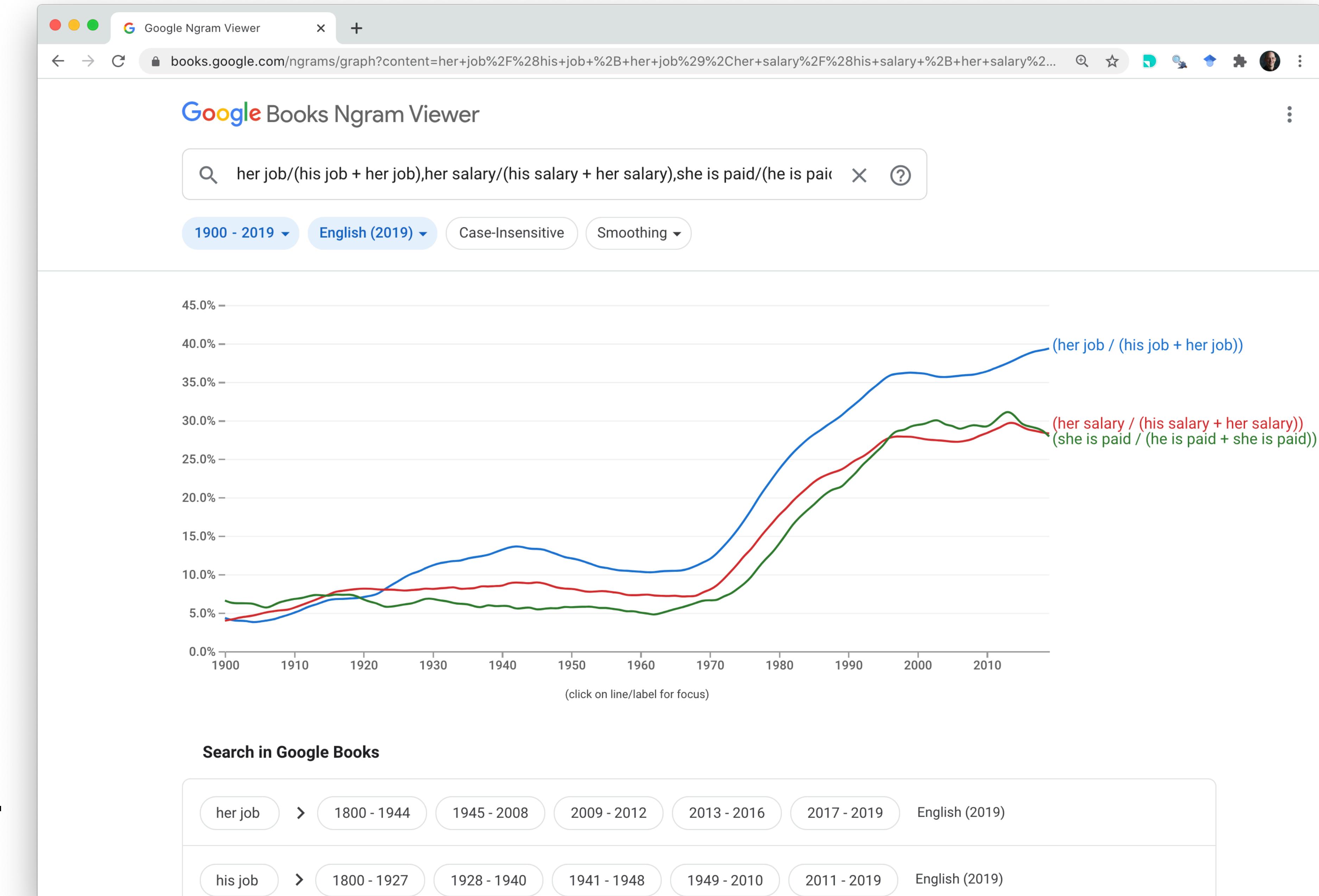


Some slides in this section have been omitted from the pdf version on purpose.

- The ngram data do have some face validity.
- People didn't talk about “jobs” until after the Industrial Revolution and labor movements.
- People don’t talk about “husbandry” as much as they used to.



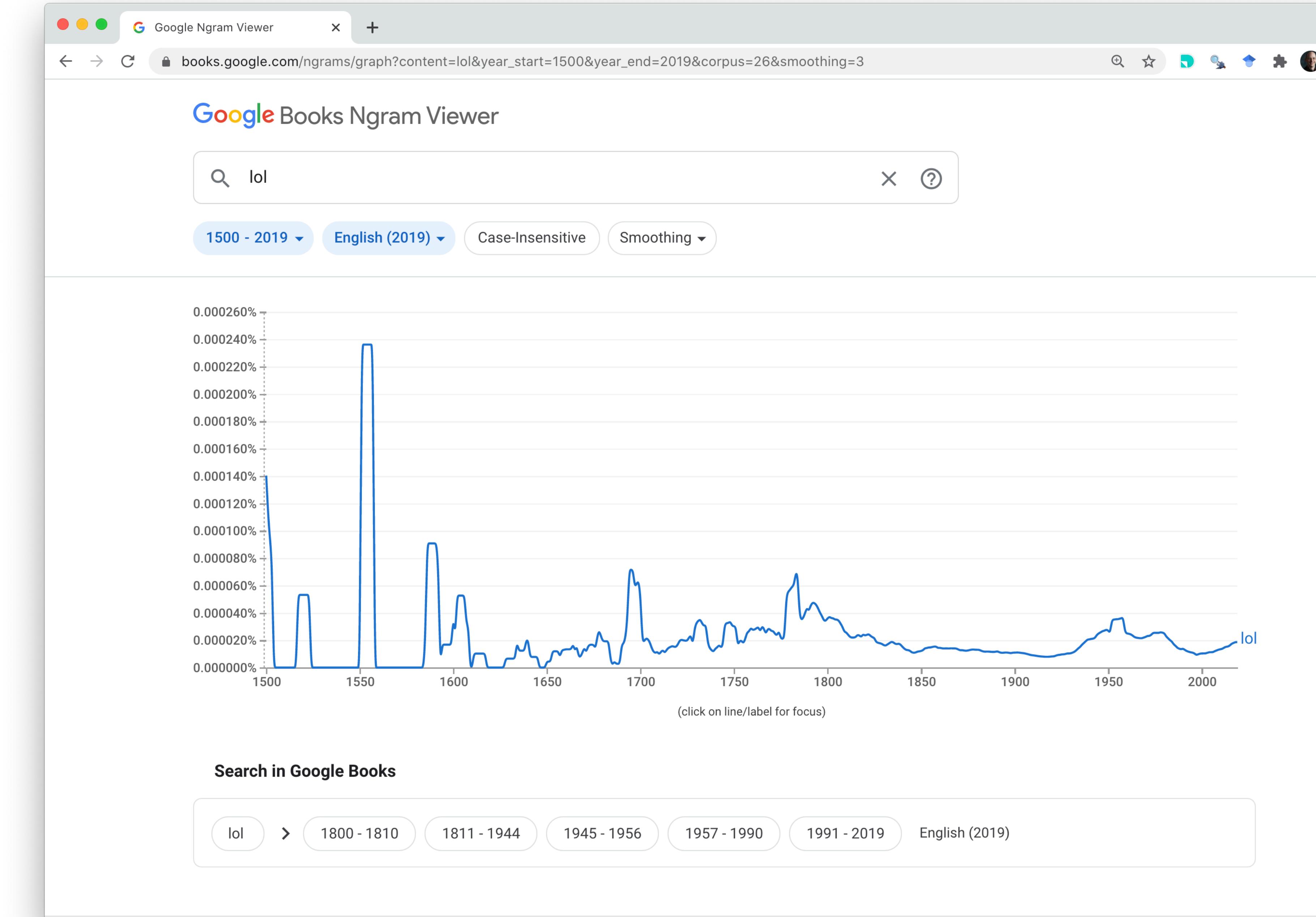
- You can do math with them!
 - The proportion of references to “her vs. his job”, “her vs. his salary”, “she vs. he is paid” is consistent and mirrors in its timing the cultural phenomenon of women moving into the workforce



- But ...

- How about how texting and online communication has changed language?

- Here we see that “lol” peaked around 1600.



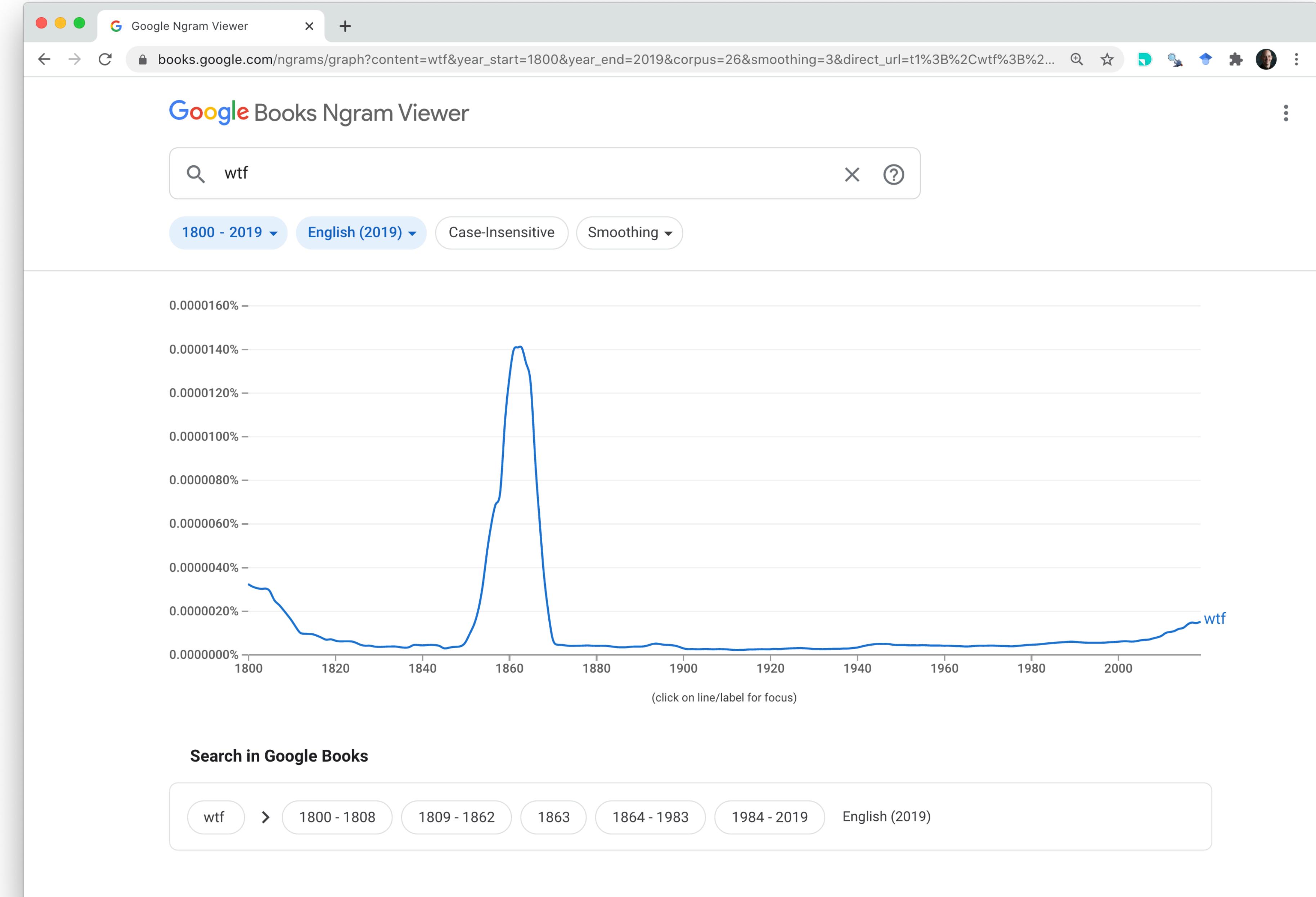
- WTF?

- Speaking of which

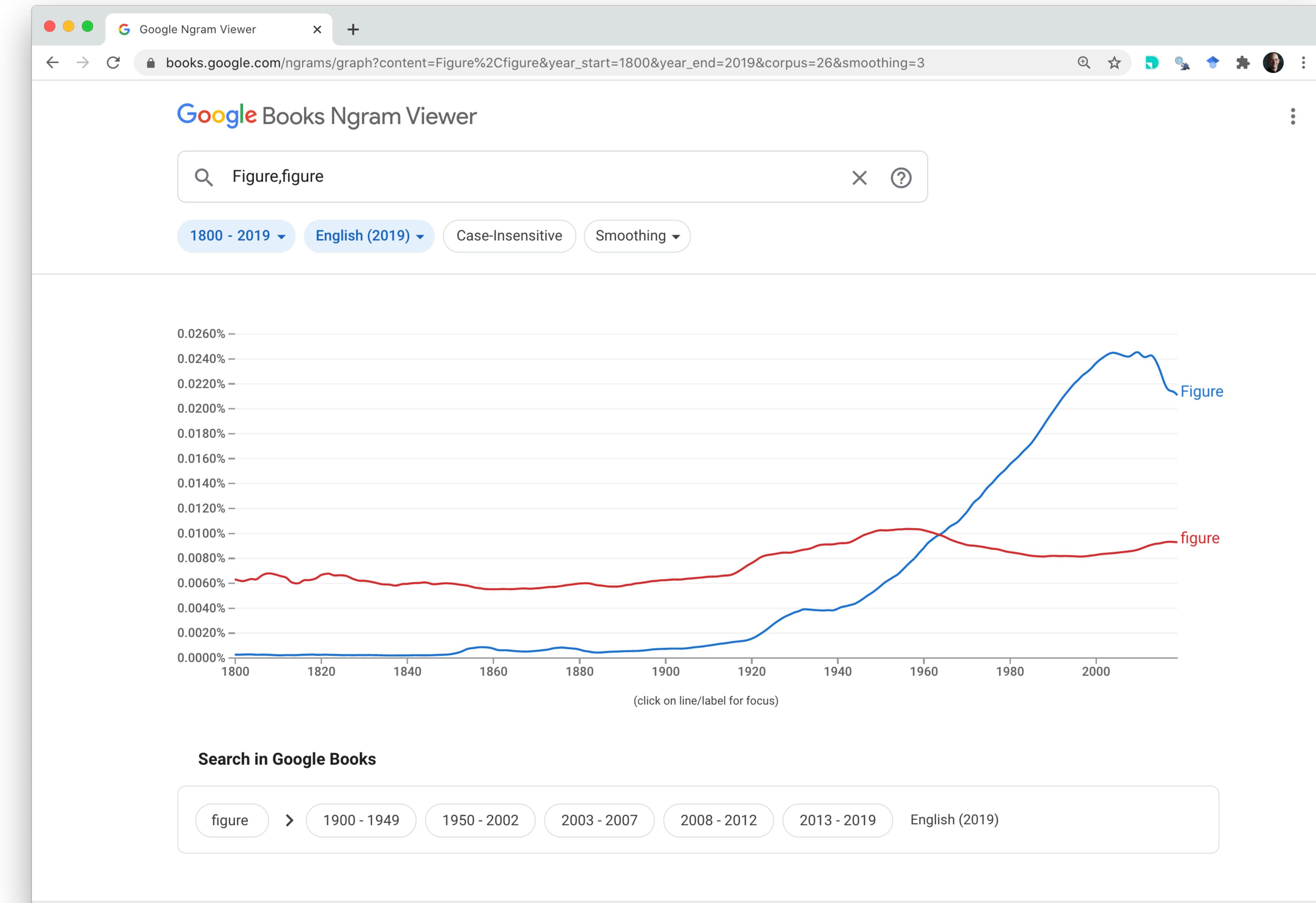
...

- “wtf” peaked around 1860.

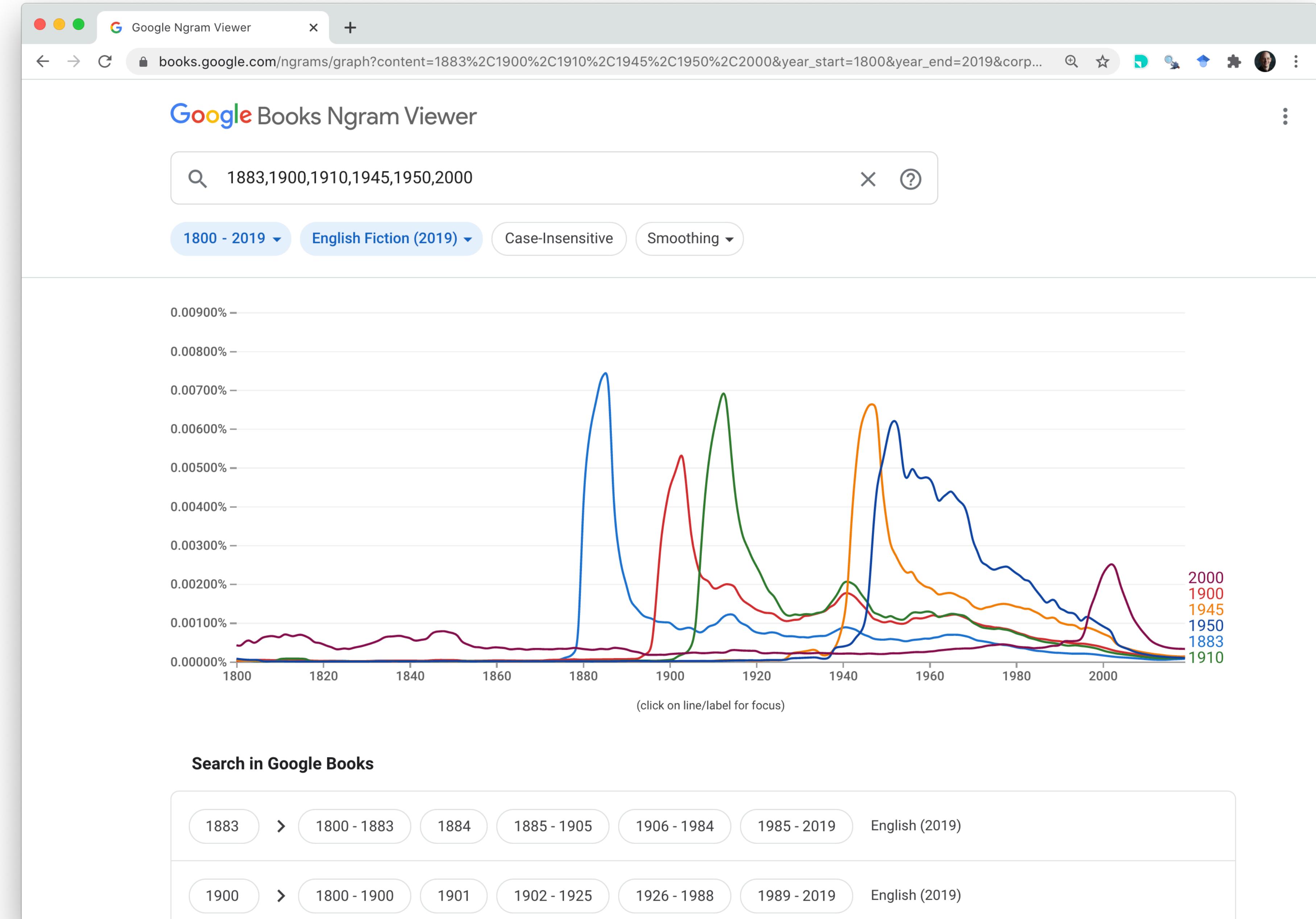
- WTAF?



- Why is “Figure” so different from “figure”?



- Plus, they cherry-picked the years they showed.
- 1900 and 1945, for example, have lingered in “cultural memory” and have a different pattern.



Semantic change - how have we changed what words mean?

Diachronic Word Embeddings Reveal Statistical Laws of Semantic Change

William L. Hamilton, Jure Leskovec, Dan Jurafsky

Department of Computer Science, Stanford University, Stanford CA, 94305
wleif, jure, jurafsky@stanford.edu

**HistWords:
Word Embeddings for Historical Text**

William L. Hamilton, Jure Leskovec, Dan Jurafsky

<https://nlp.stanford.edu/projects/histwords/>

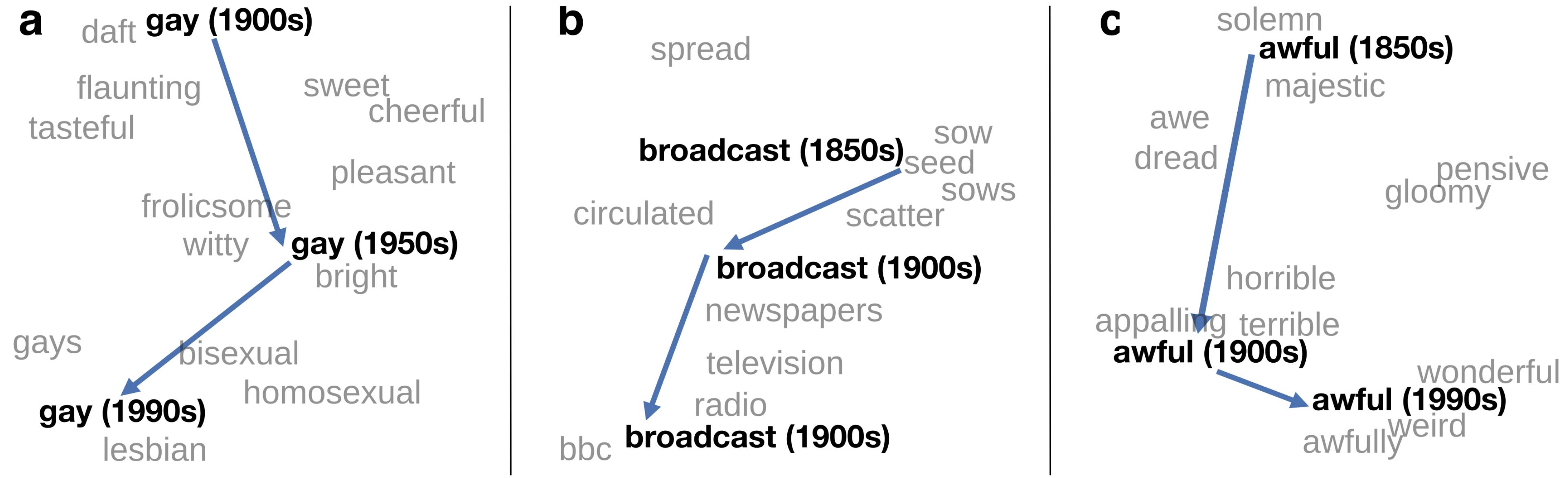


Figure 1: Two-dimensional visualization of semantic change in English using SGNS vectors.² **a**, The word *gay* shifted from meaning “cheerful” or “frolicsome” to referring to homosexuality. **b**, In the early 20th century *broadcast* referred to “casting out seeds”; with the rise of television and radio its meaning shifted to “transmitting signals”. **c**, *Awful* underwent a process of pejoration, as it shifted from meaning “full of awe” to meaning “terrible or appalling” (Simpson et al., 1989).



We use orthogonal Procrustes to align the learned low-dimensional embeddings. Defining $\mathbf{W}^{(t)} \in \mathbb{R}^{d \times |\mathcal{V}|}$ as the matrix of word embeddings learned at year t , we align across time-periods while preserving cosine similarities by optimizing:

$$\mathbf{R}^{(t)} = \arg \min_{\mathbf{Q}^\top \mathbf{Q} = \mathbf{I}} \|\mathbf{W}^{(t)} \mathbf{Q} - \mathbf{W}^{(t+1)}\|_F, \quad (4)$$

with $\mathbf{R}^{(t)} \in \mathbb{R}^{d \times d}$. The solution corresponds to the best rotational alignment and can be obtained efficiently using an application of SVD (Schönemann, 1966).

Pair-wise similarity time-series Measuring how the cosine-similarity between pairs of words changes over time allows us to test hypotheses about specific linguistic or cultural shifts in a controlled manner. We quantify shifts by computing the similarity time-series

$$s^{(t)}(w_i, w_j) = \text{cos-sim}(\mathbf{w}_i^{(t)}, \mathbf{w}_j^{(t)}) \quad (5)$$

between two words w_i and w_j over a time-period $(t, \dots, t + \Delta)$. We then measure the Spearman correlation (ρ) of this series against time, which allows us to assess the magnitude and significance of pairwise similarity shifts; since the Spearman correlation is non-parametric, this measure

Word	Moving towards	Moving away	Shift start	Source
gay	homosexual, lesbian	happy, showy	ca 1920	(Kulkarni et al., 2014)
fatal	illness, lethal	fate, inevitable	<1800	(Jatowt and Duh, 2014)
awful	disgusting, mess	impressive, majestic	<1800	(Simpson et al., 1989)
nice	pleasant, lovely	refined, dainty	ca 1900	(Wijaya and Yeniterzi, 2011)
broadcast	transmit, radio	scatter, seed	ca 1920	(Jeffers and Lehiste, 1979)
monitor	display, screen	—	ca 1930	(Simpson et al., 1989)
record	tape, album	—	ca 1920	(Kulkarni et al., 2014)
guy	fellow, man	—	ca 1850	(Wijaya and Yeniterzi, 2011)
call	phone, message	—	ca 1890	(Simpson et al., 1989)

Table 2: Set of attested historical shifts used to evaluate the methods. The examples are taken from previous works on semantic change and from the Oxford English Dictionary (OED), e.g. using ‘obsolete’ tags. The shift start points were estimated using attestation dates in the OED. The first six examples are words that shifted dramatically in meaning while the remaining four are words that acquired new meanings (while potentially also keeping their old ones).

Method	Corpus	% Correct	% Sig.
PPMI	ENGALL	96.9	84.4
	COHA	100.0	88.0
SVD	ENGALL	100.0	90.6
	COHA	100.0	96.0
SGNS	ENGALL	100.0	93.8
	COHA	100.0	72.0

Table 3: Performance on detection task, i.e. ability to capture the attested shifts from Table 2. SGNS and SVD capture the correct directionality of the shifts in all cases (%Correct), e.g., *gay* becomes more similar to *homosexual*, but there are differences in whether the methods deem the shifts to be statistically significant at the $p < 0.05$ level (%Sig).

Measuring semantic displacement After aligning the embeddings for individual time-periods, we can use the aligned word vectors to compute the semantic displacement that a word has undergone during a certain time-period. In particular, we can directly compute the cosine-distance between a word’s representation for different time-periods, i.e. $\text{cos-dist}(\mathbf{w}_t, \mathbf{w}_{t+\Delta})$, as a measure of semantic change. We can also use this measure to quantify ‘rates’ of semantic change for different words by looking at the displacement between consecutive time-points.

Method	Top-10 words that changed from 1900s to 1990s
PPMI	<u>know</u> , <u>got</u> , <u>would</u> , <u>decided</u> , <u>think</u> , <u>stop</u> , <u>remember</u> , started , <u>must</u> , <u>wanted</u>
SVD	harry, headed , <u>calls</u> , <u>gay</u> , <u>wherever</u> , <u>male</u> , actually , <u>special</u> , <u>cover</u> , <u>naturally</u>
SGNS	wanting , <u>gay</u> , <u>check</u> , <u>starting</u> , major , <u>actually</u> , <u>touching</u> , harry, headed , <u>romance</u>

Table 4: Top-10 English words with the highest semantic displacement values between the 1900s and 1990s. Bolded entries correspond to real semantic shifts, as deemed by examining the literature and their nearest neighbors; for example, *headed* shifted from primarily referring to the “top of a body/entity” to referring to “a direction of travel.” Underlined entries are borderline cases that are largely due to global genre/discourse shifts; for example, *male* has not changed in meaning, but its usage in discussions of “gender equality” is relatively new. Finally, unmarked entries are clear corpus artifacts; for example, *special*, *cover*, and *romance* are artifacts from the covers of fiction books occasionally including advertisements etc.

Word	Language	Nearest-neighbors in 1900s	Nearest-neighbors in 1990s
wanting	English	lacking, deficient, lacked, lack, needed	wanted, something, wishing, anything, anybody
asile	French	refuge, asiles, hospice, vieillards, infirmerie	demandeurs, refuge, hospice, visas, admission
widerstand	German	scheiterte, volt, stromstärke, leisten, brechen	opposition, verfolgung, nationalsozialistische, nationalsozialismus, kollaboration

Table 5: Example words that changed dramatically in meaning in three languages, discovered using SGNS embeddings. The examples were selected from the top-10 most-changed lists between 1900s and 1990s as in Table 4. In English, *wanting* underwent subjectification and shifted from meaning “lacking” to referring to subjective “desire”, as in “the education system is wanting” (1900s) vs. “I’ve been wanting to tell you” (1990s). In French *asile* (“asylum”) shifted from primarily referring to “hospitals, or infirmaries” to also referring to “asylum seekers, or refugees”. Finally, in German *Widerstand* (“resistance”) gained a formal meaning as referring to the local German resistance to Nazism during World War II.

<https://www.r-bloggers.com/2019/04/historical-word-embeddings-lexical-semantic-change/>

 R-bloggers

R news and t

HOME ABOUT RSS ADD YOUR BLOG! LEARN R R JOBS CONTACT US

historical word embeddings & lexical semantic change

Posted on April 13, 2019 by Jason Timm in R bloggers | 0 Comments

[1808,1833]	value	ppm	[1833,1858]	value	ppm	[1858,1883]	value	ppm	[1883,1908]	value	ppm
GRASP	1.00	19.8	GRASP	1.00	31.8	GRASP	1.00	30.8	GRASP	1.00	37.4
HOLD	0.38	406.3	HAND	0.40	1192.0	OPERATOR	0.38	8.2	COMPREHEND	0.56	36.8
PISTOL	0.37	14.6	SCEPTRE	0.40	17.9	GRASPED	0.35	10.8	UNDERSTAND	0.47	451.9
HAND	0.37	1065.8	COMPREHEND	0.35	51.5	COMPREHEND	0.34	39.9	GRIP	0.41	7.9
SPEAR	0.35	11.3	FIRM	0.35	96.2	HAND	0.34	1201.3	REALIZE	0.41	57.5
RELENTLESS	0.35	1.6	MIGHTY	0.34	81.6	HEARTY	0.33	27.6	CATCH	0.40	71.9
BIG	0.33	24.0	GRASPED	0.33	8.9	FINGERS	0.32	62.4	SIGNIFICANCE	0.39	71.7
SATISFY	0.31	83.7	GRASPING	0.33	4.9	FATHOM	0.32	3.7	REALISE	0.38	17.1
HAMMER	0.30	4.6	COMPREHENSIVE	0.31	34.4	HOLD	0.32	476.9	FATHOM	0.38	3.6
SLIP	0.30	18.2	COMPREHENSION	0.31	18.4	REACH	0.32	230.9	DISCOVER	0.37	117.8

[1808,1833]	value	ppm	[1833,1858]	value	ppm	[1858,1883]	value	ppm	[1883,1908]	value	ppm
COMMUNICATE	1.00	89.4	COMMUNICATE	1.00	86.8	COMMUNICATE	1.00	53.0	COMMUNICATE	1.00	45.5
COMMUNICATED	0.38	120.8	SEND	0.51	355.4	INFORM	0.50	81.8	INFORM	0.57	65.1
ARDENTLY	0.37	6.0	TRANSMIT	0.45	19.4	SEND	0.47	304.9	CONSULT	0.51	28.9
CONVEY	0.35	86.0	PUBLISH	0.45	33.8	TRANSMIT	0.43	14.6	SEND	0.46	281.1
COMPLY	0.35	83.9	INFORM	0.44	108.1	COMMUNICATED	0.43	73.7	ACQUAINT	0.44	13.7
LEARN	0.35	205.4	ACCOMPANY	0.44	58.1	IMPART	0.41	16.3	CONVEY	0.43	76.6
INFORM	0.34	131.4	GIVE	0.43	2146.6	ACCOMPANY	0.39	45.8	CONFER	0.42	27.3
GRAPPLE	0.34	2.1	WRITE	0.43	288.7	EXCELLENCY	0.38	37.4	IMPART	0.40	11.4
CONFER	0.32	36.2	DELIVER	0.41	103.3	CONVEY	0.38	84.1	PUBLISH	0.40	28.4
AVAIL	0.32	104.8	COMPLY	0.41	65.5	CONFER	0.37	32.4	EXCELLENCY	0.39	25.3

[1908,1933]	value	ppm	[1933,1958]	value	ppm	[1958,1983]	value	ppm	[1983,2008]	value	ppm
GRASP	1.00	45.9	GRASP	1.00	42.6	GRASP	1.00	47.3	GRASP	1.00	46.4
COMPREHEND	0.65	22.4	COMPREHEND	0.68	19.6	COMPREHEND	0.73	21.3	COMPREHEND	0.71	24.3
UNDERSTAND	0.63	471.6	UNDERSTAND	0.65	493.4	UNDERSTAND	0.63	601.1	UNDERSTAND	0.63	779.1
REALISE	0.60	31.0	APPRECIATE	0.61	55.2	APPRECIATE	0.59	63.3	APPRECIATE	0.60	70.9
REALIZE	0.59	136.5	PERCEIVE	0.61	41.9	REALISE	0.54	19.1	RECOGNIZE	0.55	167.9
APPRECIATE	0.56	54.3	REALISE	0.56	24.3	DETECT	0.52	30.6	REALISE	0.55	20.4
PERCEIVE	0.48	51.5	REALIZE	0.55	191.8	PERCEIVE	0.51	39.3	REALIZE	0.53	219.1
COPE	0.46	26.3	COPE	0.47	44.5	REALIZE	0.49	159.9	RECOGNISE	0.49	18.9
COMPREHENSION	0.43	17.1	GRASPED	0.43	10.3	RECOGNIZE	0.49	158.0	DISTINGUISH	0.49	164.4
CONCEIVE	0.43	66.5	RECOGNIZE	0.43	139.9	GRASPED	0.47	9.9	DETECT	0.49	47.9

[1908,1933]	value	ppm	[1933,1958]	value	ppm	[1958,1983]	value	ppm	[1983,2008]	value	ppm
COMMUNICATE	1.00	30.2	COMMUNICATE	1.00	27.0	COMMUNICATE	1.00	42.5	COMMUNICATE	1.00	66.6
COMMUNICATED	0.52	30.9	INFORM	0.57	50.5	ADAPT	0.49	34.8	COMPETE	0.46	56.3
TRANSMIT	0.52	13.8	ACQUAINT	0.52	14.1	UNDERSTAND	0.46	601.1	INTERACT	0.44	23.2
SEND	0.45	240.7	TRANSMIT	0.47	14.7	ADJUST	0.45	31.3	LEARN	0.44	420.2
IMPART	0.45	10.2	SEND	0.42	225.2	IDENTIFY	0.45	91.3	CONVEY	0.43	45.0
INFORM	0.44	53.8	COMMUNICATED	0.41	22.1	COPE	0.44	83.2	MOTIVATE	0.39	1.7
CONSULT	0.44	20.8	EXPRESS	0.40	218.3	CONVINCE	0.44	53.4	MANAGE	0.39	43.7
CONVEY	0.41	43.5	CONSULT	0.40	24.8	TRANSMIT	0.44	11.1	USER	0.39	68.2
ACQUAINT	0.40	15.1	IMPART	0.39	5.9	DISLodge	0.43	1.7	MANIPULATE	0.38	6.1
PUBLISH	0.39	23.2	ANNOUNCE	0.38	15.7	RESPOND	0.42	73.8	UTILIZE	0.38	6.7

Another approach - model dynamics directly

Dynamic Word Embeddings for Evolving Semantic Discovery

Zijun Yao

Rutgers University

zijun.yao@rutgers.edu

Yifan Sun

Technicolor Research

yifan.sun@technicolor.com

Weicong Ding

Amazon

20008005dwc@gmail.com

Nikhil Rao

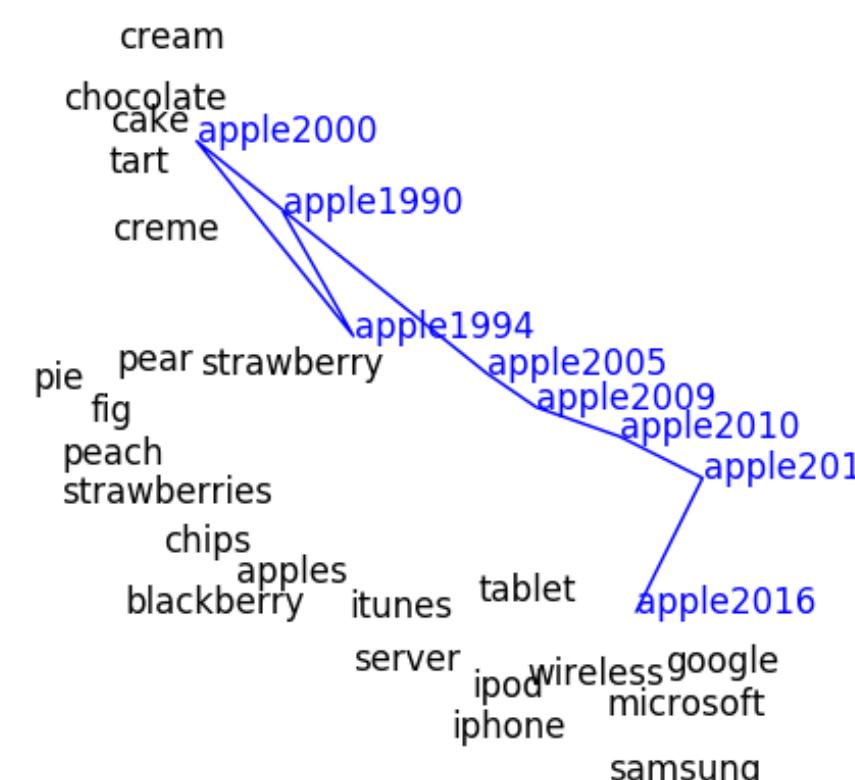
Amazon

nikhilrao86@gmail.com

Hui Xiong

Rutgers University

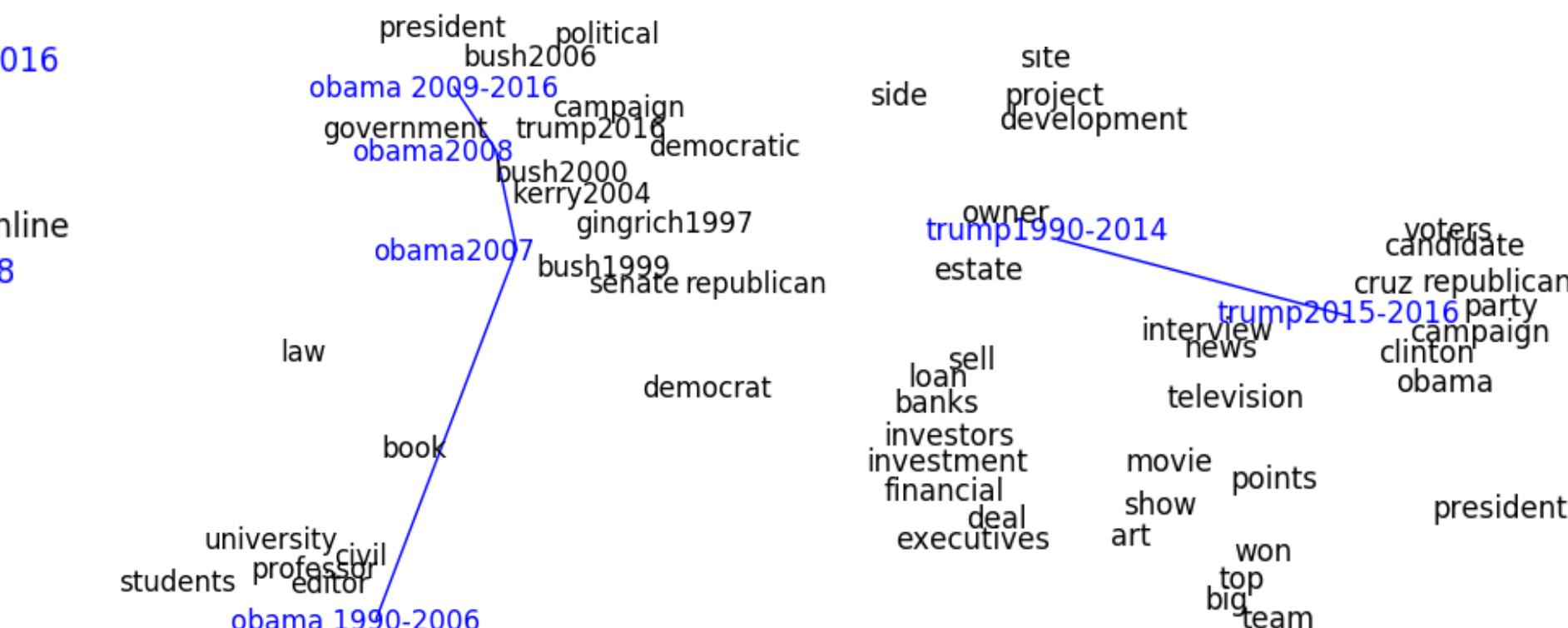
hxiong@rutgers.edu



(a) apple



(b) amazon



(c) obama

(d) trump

Figure 1: Trajectories of brand names and people through time: apple, amazon, obama, and trump.

Table 2: “Who governed?” The closest word to obama at year 2016 (role as president of United State) and blasio at year 2015 (role as mayor of New York City (NYC)). The stars indicate incorrect answers.

Table 1: Equivalent technologies through time: iphone, twitter, and mp3.

Query	iphone, 2012	twitter, 2012	mp3, 2000
90-94	desktop, pc, dos, macintosh, software	broadcast, cnn, bulletin, tv, radio, messages, correspondents	stereo, disk, disks, audio
95-96			mp3
97		chat, messages, emails, web	
98-02	pc		napster
03			mp3
04			
05-06	ipod		
07-08		blog, posted	
09-12	iphone		itunes, downloaded
13-16	smartphone, iphone	twitter	

Question	US president	NYC mayor
Query	obama, 2016	blasio, 2015
90-92	bush	dinkins
93	clinton	giuliani
94-00		
01	bush	bloomberg
02-05		n/a*
06		
07		bloomberg
08		
09-10	obama	cuomo*
11		bloomberg
12		
13-16		blasio

Table 3: “Who was the ATP No.1 ranked male player?” The closest word to nadal at year 2010 for each year is listed. The correct answer is based on ATP year-end ranking and are bolded in the table.

year	1990	1991	1992	1993
word	edberg	lendl	sampras	sampras
1994	1995	1996	1997	1998
sampras	sampras	ivanisevic	sampras	sampras
1999	2000	2001	2002	2003
sampras	sampras	agassi	capriati	roddick
2004	2005	2006	2007	2008
federer	federer	roddick	federer	nadal
2009	2010	2011	2012	2013
federer	nadal	djokovic	federer	federer
2014	2015			
		djokovic		

SemEval-2020 Task 1: Unsupervised Lexical Semantic Change Detection

**Dominik Schlechtweg,[♣] Barbara McGillivray,^{◊,♡} Simon Hengchen,^{♣*}
Haim Dubossarsky,[♡] Nina Tahmasebi[♣]**

semeval2020lexicalsemanticchange@turing.ac.uk

[♣]University of Stuttgart, [◊]The Alan Turing Institute, [♡]University of Cambridge
[♣]University of Gothenburg

Abstract

Lexical Semantic Change detection, i.e., the task of identifying words that change meaning over time, is a very active research area, with applications in NLP, lexicography, and linguistics. Evaluation is currently the most pressing problem in Lexical Semantic Change detection, as no gold standards are available to the community, which hinders progress. We present the results of the first shared task that addresses this gap by providing researchers with an evaluation framework and manually annotated, high-quality datasets for English, German, Latin, and Swedish. 33 teams submitted 186 systems, which were evaluated on two subtasks.

Senses	C_1			C_2		
	chamber	biology	phone	chamber	biology	phone
# uses	12	18	0	4	11	18

Table 1: An example of a sense frequency distribution for the word *cell* in C_1 and C_2 .

different model architectures to be applied to it, widening the range of possible participants. Participants were asked to solve two subtasks:

Subtask 1 Binary classification: for a set of target words, decide which words lost or gained sense(s) between C_1 and C_2 , and which ones did not.

Subtask 2 Ranking: rank a set of target words according to their degree of LSC between C_1 and C_2 .

6 Participating Systems

Thirty-three teams participated in the task, totaling 53 members. The teams submitted a total of 186 submissions. Given the large number of teams, we provide a summary of the systems in the body of this paper. A more detailed description of each participating system for which a paper was submitted is available in Appendix B. We also encourage the reader to read the full system description papers.

Participating models can be described as a combination of (i) a semantic representation, (ii) an alignment technique and (iii) a change measure. Semantic representations are mainly average embeddings (*type* embeddings) and contextualized embeddings (*token* embeddings). Token embeddings are often combined with a clustering algorithm such as K-means, affinity propagation (Frey and Dueck, 2007), (H)DBSCAN, GMM, or agglomerative clustering. One team uses a graph-based semantic network, one a topic model and several teams also propose ensemble models. Alignment techniques include Orthogonal Procrustes (Hamilton et al., 2016, OP), Vector Initialization (Kim et al., 2014, VI), versions of Temporal Referencing (Dubossarsky et al., 2019, TR), and Canonical Correlation Analysis (CCA). A variety of change measures are applied, including Cosine Distance (CD), Euclidean Distance (ED), Local Neighborhood Distance (LND), Kullback-Leibler Divergence (KLD), mean/standard deviation of co-occurrence vectors, or cluster frequency. Table 5 shows the type of system for every team’s best submission for both subtasks.

Remarkably, all the top performing systems use static-type embedding models, and differ only in terms of their solutions to the alignment problem (Canonical Correlation Analysis, Orthogonal Procrustes, or Temporal Referencing). Interestingly, the top systems refine their models using one or more of the following steps: a) computing additional features from the embedding space; b) combining scores from different models (or extracted features) using ensemble models; c) choosing a threshold for changed words based on a distribution of change scores. We conjecture that these additional (and sometimes very original) post-processing steps are crucial for these systems' success. We now briefly describe the top performing systems in terms of these three steps (for further details please see Appendix B). **UWB** (SGNS+CCA+CD) sets the average change score as the threshold (c). **Life-Language** (SGNS) represents words according to their distances to a set of stable pivot words in two unaligned spaces, and compares their divergence relative to a distribution of change scores obtained from unstable pivot words (a+c). **RPI-Trust** (SGNS+OP) extract features (a word's cosine distance, change of distances to its nearest-neighbours and change in frequency), transform each word's feature to a CDF score, and averages these probabilities (a+b+c). **Jiaxin & Jinan** (SGNS+TR+CD) fits the empirical cosine distance change scores to a Gamma Quantile Threshold, and sets the 75% quantile as the threshold (c). **UG_Student_Intern** (SGNS+OP) measures change using Euclidean distance instead of cosine distance. **cs2020** uses SGNS+OP+CD only

Type versus token embeddings Tables 5 and 6 illustrate the gap in performance between type-based embedding models and the token-based ones. Out of the best 10 systems in Subtask 1/Subtask 2, 7/8 systems are based on type embeddings compared to only 2/2 systems that are based on token embeddings (same holds for each language individually). Contrary to the recent success of token embeddings (Peters et al., 2018) and to commonly held view that contextual embeddings “do everything better”, they are overwhelmingly outperformed by type embeddings in our task. This is most surprising for Subtask 1, because type embeddings do not distinguish between different senses, while token embeddings do. We suggest several possible reasons for these surprising results. The first is the fact that contextual embedding is a recent technology, and as such lacks proper usage conventions. For example, it is not clear whether a model should create an average token representation based on individual instances (and if so, which layers should be averaged), or if it should use clustering of individual instances instead (and if so, what type of clustering algorithm etc.). A second reason may be related to the fact that contextual models are pretrained and cannot exclusively be trained on the relevant historical resources (in contrast to type embeddings). As such, they carry additional, and possibly irrelevant, information that may mask true diachronic changes. The results may also be related to the specific preprocessing we applied to the corpora: (i) Only restricted context is available to the models as a result of the sentence shuffling. Usually, token-based models take more context into account than just the immediate sentence (Martinc et al., 2020). (ii) The corpora were lemmatized, while token-based models usually take the raw sentence as input. In order to make the input more suitable for token-based models, we also provide the raw corpora after the evaluation phase and will publish the annotated uses of the target words with additional context.¹⁷

The influence of frequency In prior work, the predictions of many systems have been shown to be inherently biased towards word frequency, either as a consequence of an increasing sampling error with lower frequency (Dubossarsky et al., 2017) or by directly relying on frequency-related variables (Schlechtweg et al., 2017; Schlechtweg et al., 2019). We have controlled for frequency when selecting target words (recall Table 4) in order to test model performance when frequency is not an indicating factor. Despite the controlled test sets we observe strong frequency biases for the individual models as illustrated for Swedish in Figure 3.¹⁸ Models rather correlate negatively with the minimum frequency of target words between corpora (FRQ_m), and positively with the change in their frequency across corpora (FRQ_d). This means that models predict higher change for low-frequency words and higher change for words with strong changes in frequency. Despite their superior performance, type embeddings are more