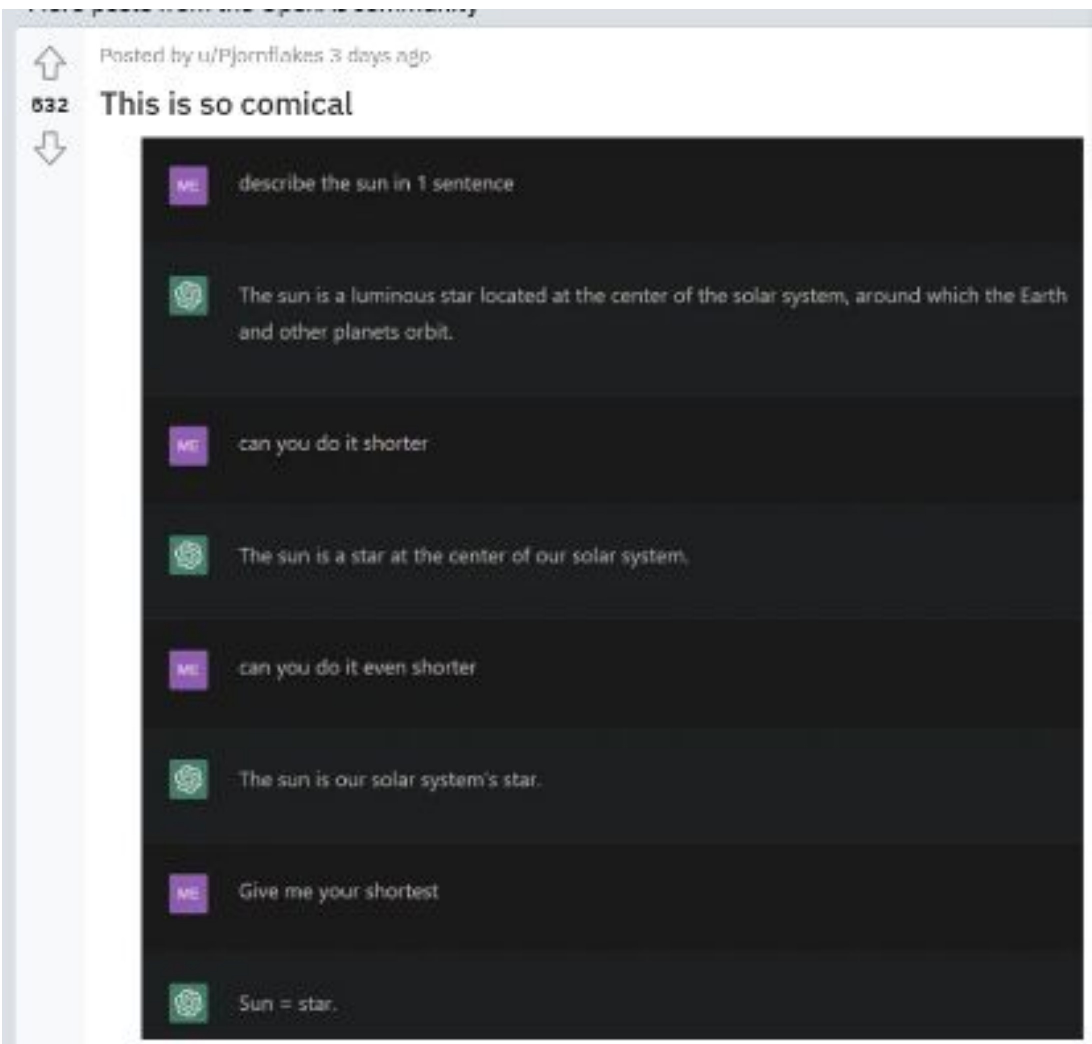
Advanced Quantitative Techniques
(Week 7)

Gregory M. Eirich
QMSS

Agenda

1. How do I get text again?
2. Running regressions on all these words
3. Word 2 vector models
4. What is natural language processing (NLP) vs. text analysis?
5. Why text analysis again?

ChatGPT as summarizer



Text cleaning without coding?

Nicole Leffer • 3rd+ + Follow •••

Tech Marketing Leader | Leveraging AI in Marketing | Growt...

4d •

Just found a new AI work hack that everyone can use (regardless of job function or industry), and it's not one I've seen anyone else talk about so I wanted to share!

We've all been here: You're on a virtual event or webinar with a HIGHLY engaged chat. The presenter asks something and the chat goes wild with responses coming in a mile a minute. It's information that would be SO helpful to have compiled in one place and know what people said, but no way you can go through it all. Oh well.

This just happened on a call I was on. The presenter asked the group "what is your favorite AI tool that's already in your marketing tech stack? Put it in the chat!" There were almost 500 people on the call, so the chat blew up fast. There were tons of answers, and many of them were duplicates of each other.

I wanted the list.

Here's how I got a cleanly compiled list with NO duplicates in 30 seconds with ChatGPT:

1. I waited for the rapid-fire chat responses to stop. Then I copied and pasted the chat from the place the entries started to the place they ended and opened ChatGPT.

2. I entered the following prompt:

"I'm on a zoom call and people were asked what AI tools are their favorites. Compile a list for the responses - remove the names and to everyone and time and only list the tools. Do not duplicate the same tool multiple times."

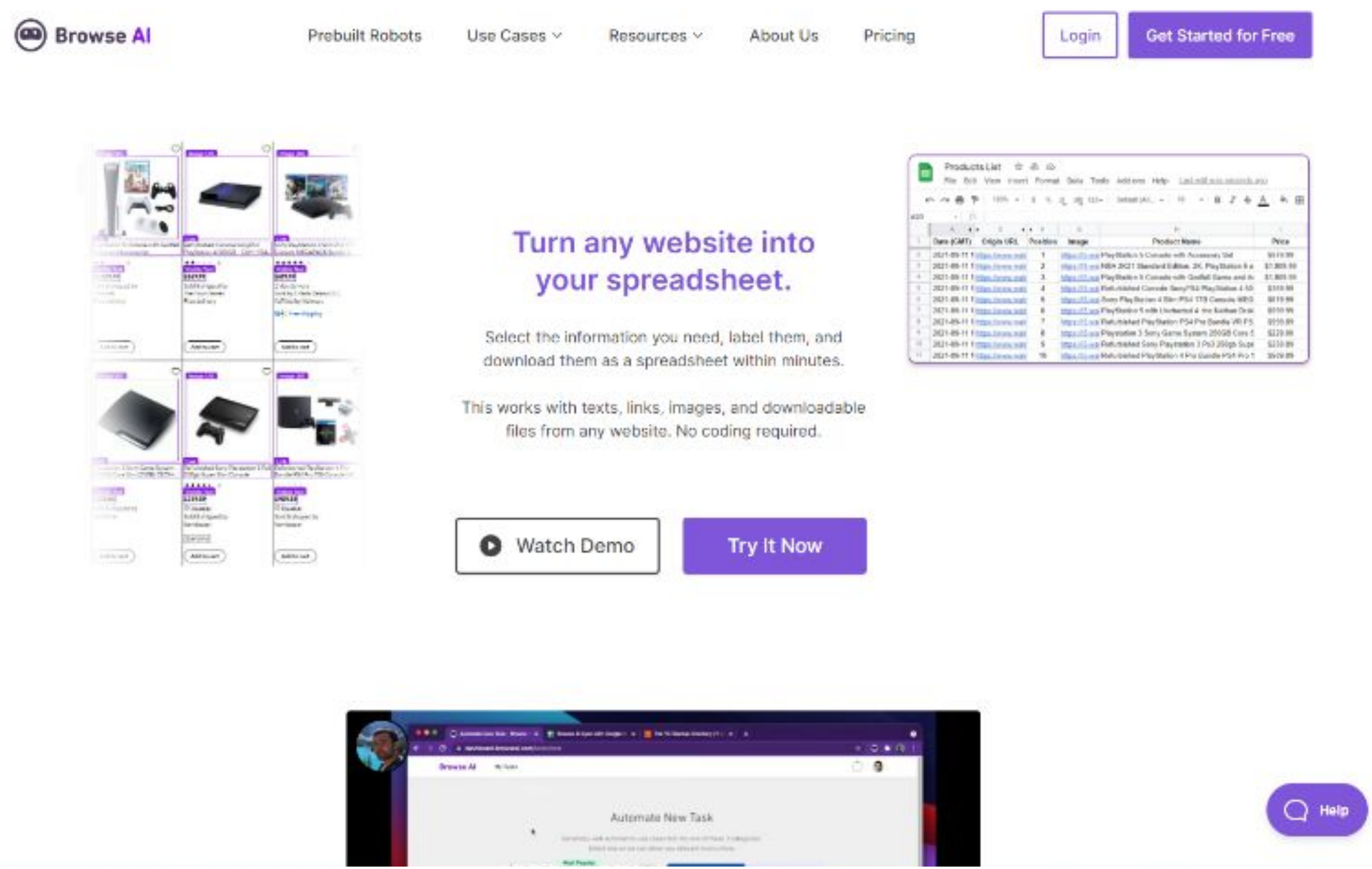
Then I pasted in the chat transcript (complete with names, the time it was sent, and the responses).

3. In 10 seconds I had a clean list of the 17 tools people had listed.

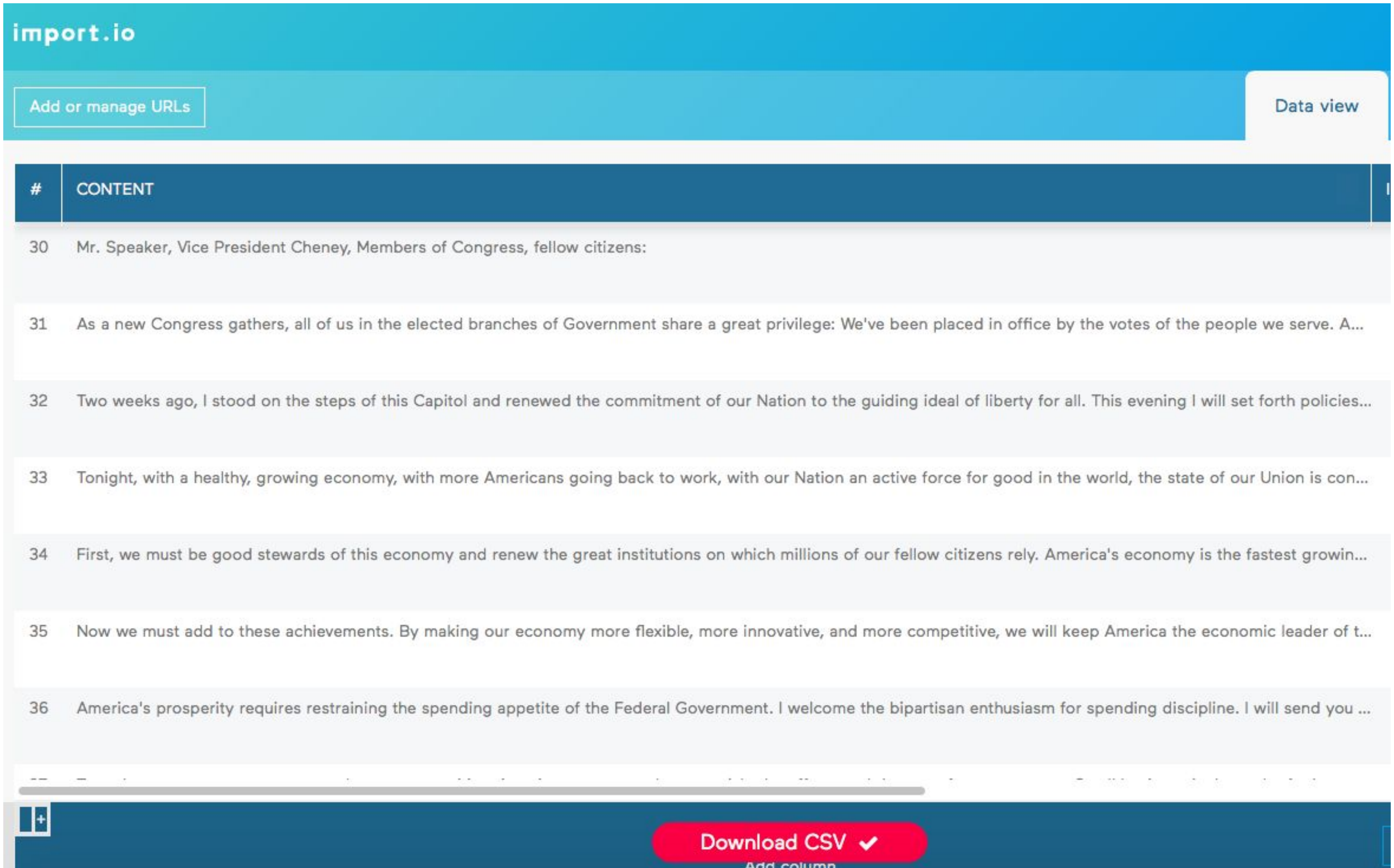
1. How do I get text again?

How do I get Bush's speech into a .csv?

One option: **no code web scraping**



Here's how to get Bush's speech into a .csv?



Another option: read it into R

Get the speech from the website

```
> thepage = readLines("http://www.presidency.ucsb.edu/ws/index.php?pid=58746")

> head(thepage)
[1] "<html>"
[2] "<head>"
[3] "<META HTTP-EQUIV=\"Content-Type\" CONTENT=\"text/html; charset=windows-1251\">"
[4] "<meta name=\"keywords\" content=\"President of the United States, presidency, 
American Presidency, American President, Public Papers of the Presidents, State of the 
Union Address, Inaugural Address, Presidents, American Presidents, George W. Bush, 
Bill Clinton, George Bush, Ronald Reagan, Jimmy Carter, Gerald Ford, Richard Nixon, 
Lyndon Johnson, John F. Kennedy. John Kennedy, Dwight Eisenhower, Harry Truman, FDR, 
Franklin Roosevelt, Presidential Elections, Presidential Rhetoric\">"
[5] "<meta name=\"description\" content=\"The American Presidency Project contains the 
most comprehensive collection of resources pertaining to the study of the President of 
the United States.  Compiled by John Woolley and Gerhard Peters\">"
[6] "<link href=\"http://www.presidency.ucsb.edu/styles/main.css\" rel=\"stylesheet\" 
type=\"text/css\">"
```

Finding what we want

I find it on line 1143

```
> thepage[1143]
[1] "   </div></div><span class=\"displaytext\">Mr. Speaker, Vice President Cheney,
Members of Congress, fellow citizens: <p>As a new Congress gathers, all of us in the
elected branches of Government share a great privilege: We've been placed in office by
the votes of the people we serve. And tonight that is a privilege we share with newly
[...omitted...] and to spread the peace that freedom brings. <p>As Franklin Roosevelt
once reminded Americans, \"Each age is a dream that is dying, or one that is coming to
birth.\" And we live in the country where the biggest dreams are born. The abolition
of slavery was only a dream until it was fulfilled. The liberation of Europe from
fascism was only a dream until it was achieved. The fall of imperial communism was
only a dream until, one day, it was accomplished. Our generation has dreams of its
own, and we also go forward with confidence. The road of providence is uneven and
unpredictable, yet we know where it leads: It leads to freedom. <p>Thank you, and may
God bless America.</span><hr noshade=\"noshade\" size=\"1\"><span
class=\"displaynotes\"><i>NOTE: The President spoke at 9:10 p.m. in the House Chamber
of the Capitol. In his remarks, he referred to senior Al Qaida associate Abu Musab Al
Zarqawi; Prime Minister Ariel Sharon of Israel; President Mahmoud Abbas (Abu Mazen) of
the Palestinian Authority; former President Saddam Hussein of Iraq; and Prime Minister
Ayad Allawi of the Iraqi Interim Government. The Office of the Press Secretary also
released a Spanish language transcript of this address.</i></span><hr
noshade=\"noshade\" size=\"1\">"
```

Finding what we want

Just get line 1143

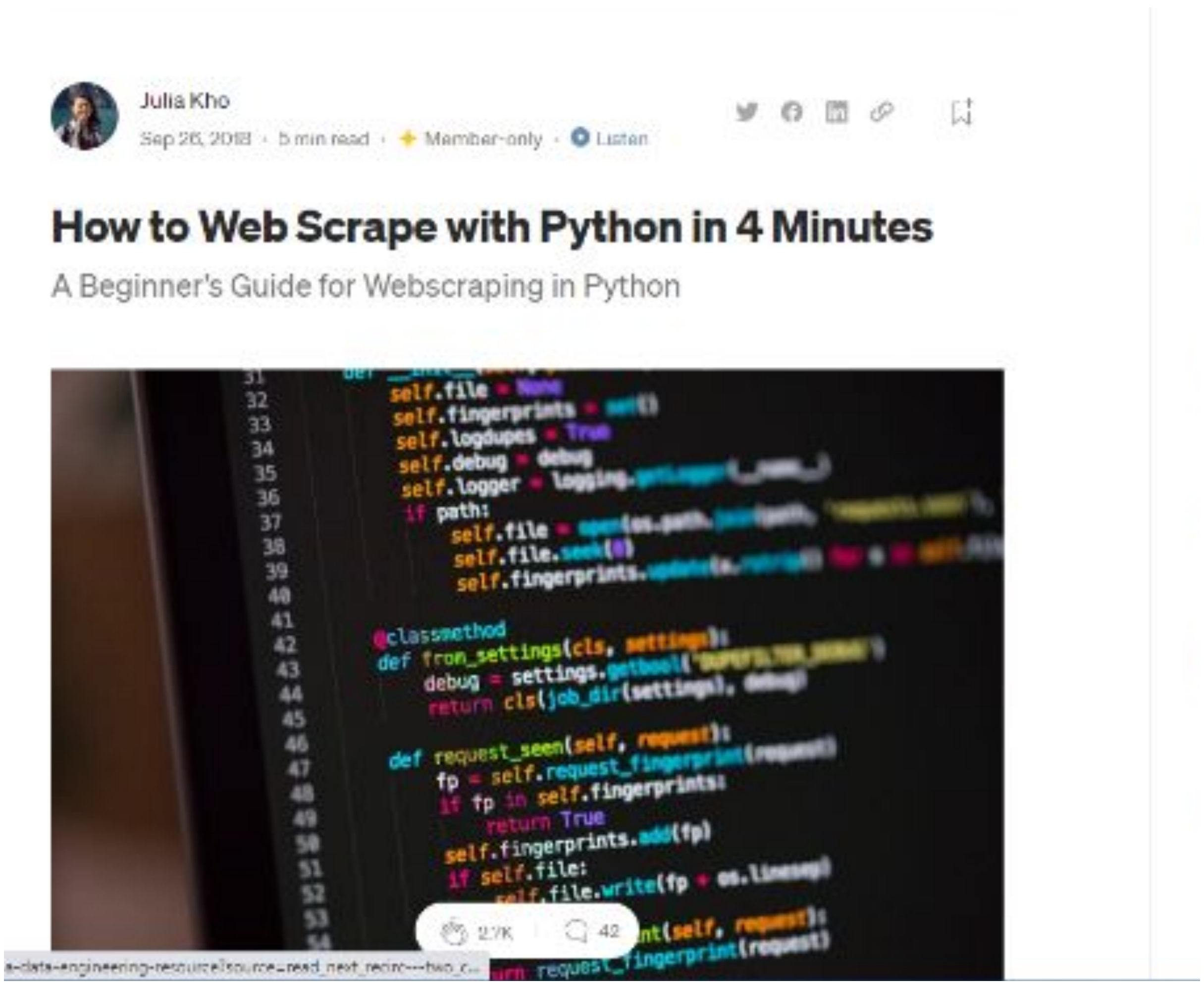
```
p = thepage[1143]

dp <- Corpus(VectorSource(p))

inspect(dp)

docs2 <- tm map(dp, function(x) stri replace all regex(x, "<.+?>", " "))
docs3 <- tm_map(docs2, function(x) stri_replace_all_fixed(x, "\t", " "))

docs4 <- tm map(docs3, PlainTextDocument)
docs5 <- tm map(docs4, stripWhitespace)
docs6 <- tm map(docs5, removeWords, stopwords("english"))
docs7 <- tm map(docs6, removePunctuation)
docs8 <- tm_map(docs7, tolower)

docs8[[1]]
```

Finding what we want

I find it on line 1143

```
> docs8[[1]]
 mr speaker vice president cheney members  congress fellow citizens as  new congress
gathers   us   elected branches  government share  great privilege weve  placed
office   votes   people  serve and tonight    privilege  share  newly elected leaders
afghanistan  palestinian territories ukraine   free  sovereign iraq two weeks ago i
stood   steps   capitol  renewed  commitment   nation   guiding ideal  liberty   this
evening i will set forth policies  advance  ideal  home  around  world tonight
healthy growing economy   americans going back  work   nation  active force  good
world  state   union  confident  strong our generation   blessed   expansion
opportunity  advances  medicine   security purchased   parents sacrifice now
[...omit…] freedom   world  reaffirmed  confidence  freedoms power  change  world we
part   great venture to extend  promise  freedom   country  renew  values  sustain
liberty   spread  peace  freedom brings as franklin roosevelt  reminded americans each
age   dream   dying  one   coming  birth and  live   country   biggest dreams  born
the abolition  slavery    dream    fulfilled the liberation  europe  fascism    dream
achieved the fall  imperial communism    dream  one day   accomplished our generation
dreams      also go forward  confidence the road  providence  uneven  unpredictable
yet  know   leads it leads  freedom thank   may god bless america note the president
spoke  910 pm   house chamber   capitol in  remarks  referred  senior al qaida
associate abu musab al zarqawi prime minister ariel sharon  israel president mahmoud
abbas abu mazen   palestinian authority former president saddam hussein  iraq  prime
minister ayad allawi   iraqi interim government the office   press secretary also
released  spanish language transcript   address
```

Similar steps in Python



Other ideas?

What have you encountered in your other work?

2. Running regressions on all these words

A bigger example

- Preoţiuc-Pietro, Daniel, et al. "Studying user income through language, behaviour and affect in social media." *PloS one* 10.9 (2015): e0138717.

The question

- Is it possible to predict someone’s income based on what they write on Twitter?

The answer

- Is it possible to predict someone’s income based on what they write on Twitter?

- Yes-- They hypothesise that income is revealed through a variety of factors, starting from the actual text posted by a user, but also via other information, such as the number of friendships, demographics (e.g. gender and age), personality, perceived intelligence, education level and expressed emotions.

How do they know people’s income?

The features used to predict income

Table 2. Description of the user level features.

(a) User profile features (Profile)	
u_1	number of followers
u_2	number of friends
u_3	number of times listed
u_4	follower/friend ratio
u_5	no. of favourites the account made
u_6	avg. number of tweets/day
u_7	total number of tweets
u_8	proportion of tweets in English
(b) User psycho-demographic features (Demo)	
d_1	gender (male, female)
d_2	age (18–70)
d_3	political (independent, conservative, liberal, unaffiliated)
d_4	intelligence (> average, average, $\leq$ average, $\gg$ average, $\ll$ average)
d_5	relationship (married, in a relationship, single, other)
d_6	ethnicity (Asian, African American, Indian, Hispanic, Other, Caucasian)
d_7	education (bachelor, graduate, high school)
d_8	religion (Christian, Jewish, Muslim, Hindu, unaffiliated, other)
d_9	children (yes, no)
d_{10}	income (below average, above average, very high)
d_{11}	life satisfaction (satisfied, dissatisfied, very satisfied, very dissatisfied, neither)
d_{12}	optimism (optimist, pessimist, extreme optimist, extreme pessimist, neither)
d_{13}	narcissism (agree strongly, agree, disagree, disagree strongly, neither)
d_{14}	excited (agree strongly, agree, disagree, disagree strongly, neither)
d_{15}	anxious (agree strongly, agree, disagree, disagree strongly, neither)

The features used to predict income

(c) User emotion features (Emo)	
e_1	proportion of tweets with positive sentiment
e_2	proportion of tweets with neutral sentiment
e_3	proportion of tweets with negative sentiment
e_4	proportion of joy tweets
e_5	proportion of sadness tweets
e_6	proportion of disgust tweets
e_7	proportion of anger tweets
e_8	proportion of surprise tweets
e_9	proportion of fear tweets
(d) Shallow textual features (Shallow)	
s_1	proportion of non-duplicate tweets
s_2	proportion of retweeted tweets
s_3	average no. of retweets/tweet
s_4	proportion of retweets done
s_5	proportion of hashtags
s_6	proportion of tweets with hashtags
s_7	proportion of tweets with @-mentions
s_8	proportion of @-replies
s_9	no. of unique @-mentions in tweets
s_{10}	proportion of tweets with links

How well can they predict?

Table 3. Prediction of income with our groups of features. Pearson correlation (left columns) and Mean Average Error (right columns) between income and our models on 10 fold cross-validation using three different regression methods: Linear regression (LR), Support Vector Machines with RBF kernel (SVM) and Gaussian Processes (GP) and sets of features described in the User Features section.

Feature set	No. Features	LR		SVM		GP	
Profile	8	.205	£11460	.331	£11033	.372	£11291
Demo	15	.278	£11126	.257	£10418	.364	£10110
Emo	9	.271	£11093	.358	£10768	.371	£10980
Shallow	10	.200	£11183	.261	£11494	.355	£11456
Topics	200	.498	£10430	.606	£9835	.608	£9621
All features (Linear ensemble)	5	.506	£10342	.614	£9652	.633	£9535

How well can they predict?

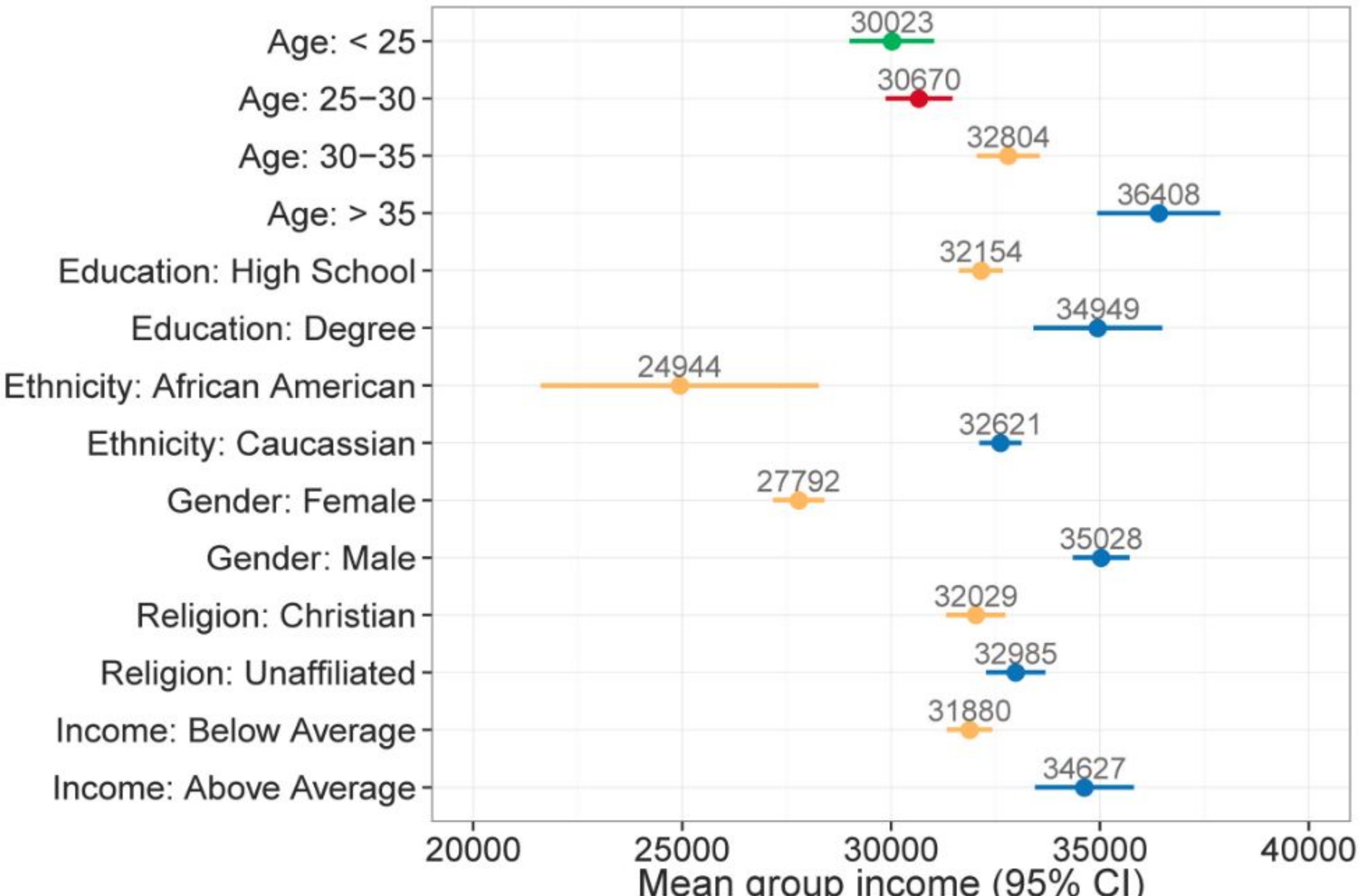


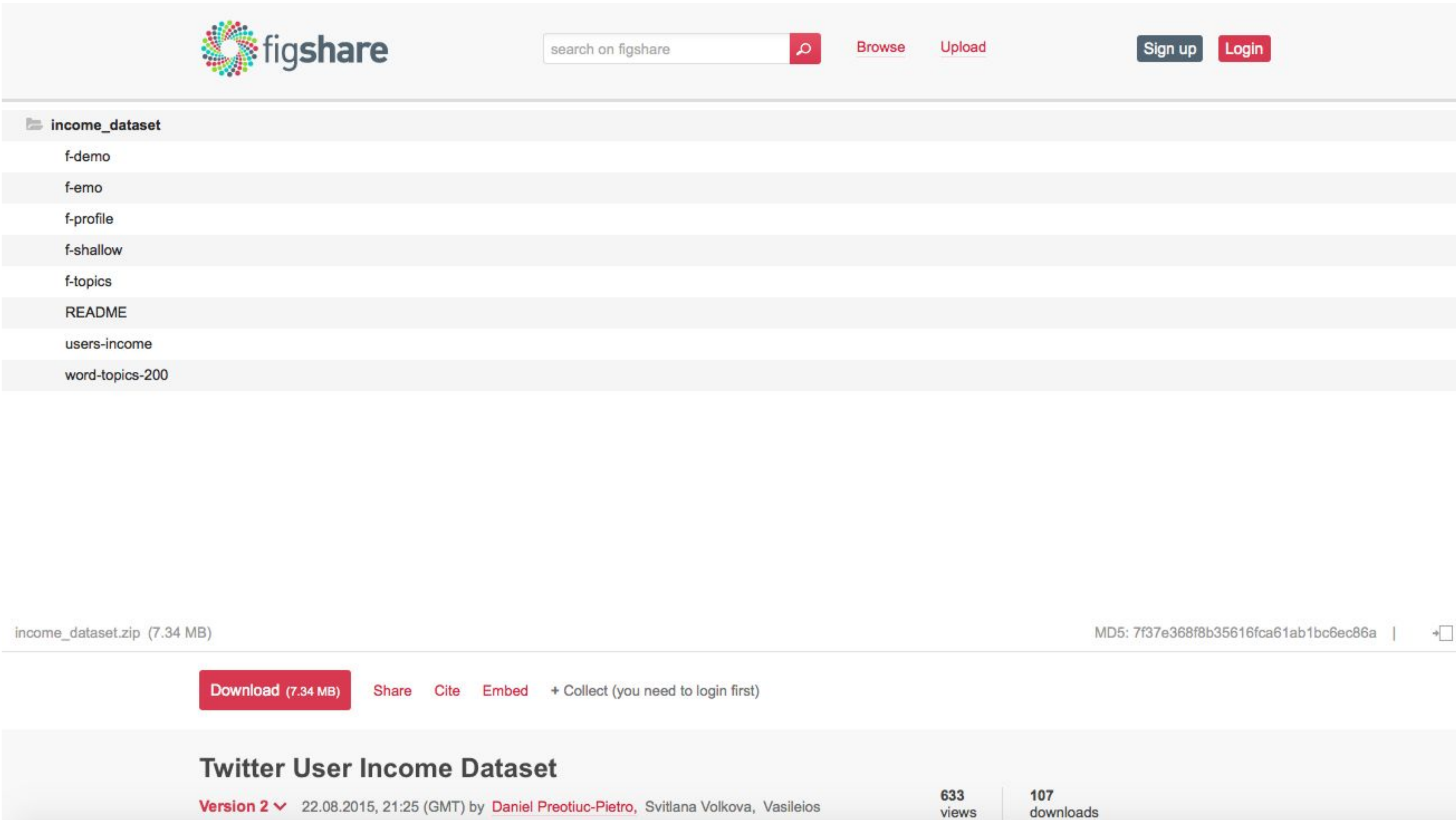
Fig 2. Mean income with confidence intervals for psycho-demographic groups. All group mean differences are statistically significant (Mann-Whitney test, $p < .001$).

What topics matter most?

Table 4. Topics, represented by top 15 words, sorted by their ARD lengthscale. Most predictive topics for income. Topic labels are manually added. Lower lengthscales (l) denote more predictive topics.

Rank	Topic #	Label	Topic	l
1	139	Politics	republican democratic gop congressional judiciary hearings abolishing oppose legislation governors congress constitutional lobbyists democrat republicans	3.10
2	163	NGOs	advocacy organization organizations advocates disadvantaged communities organisations participation outreach associations non-profit nonprofit orgs educators initiative	3.44
3	196	Web analytics / Surveys	#measure analytics #mrx #crowdsourcing crowdsourcing #socialmedia #analytics whitepaper #li metrics #roi startup #social #smm segmentation	3.68
5	124	Corporate 1	consortium institutional firm's acquisition enterprises subsidiary corp telecommunications infrastructure partnership compan aims telecom strategic mining	6.48
6	91	Corporate 2	considerations provides comprehensive cost-effective enhance advantages selecting utilizing resource essential additionally specialized benefits provide enhancing	7.44
7	107	Justice	allegations prosecution indictment alleged convicted allegation alleges accused charges extortion defendant investigated prosecutor sentencing unlawful	7.84
8	92	Link words	otherwise unless wouldn't whatever either maybe pretend anyone's assume eventually assuming or bother couldn't however	8.39
9	173	Beauty	hair comb bleached combed slicked hairs eyebrows ponytail trimmed curlers dye dyed curls waxed bangs	9.75
10	40	Sport shows	first-ever roundup sport's round-up rundown poised previewing spotlight thursday's com's long-running joins concludes prepares observer	10.57
11	99	Swearing	messed f'd picking effed cracking f*cked hooking tearing catching lighten picked cracks ganging warmed fudged	11.09

You can recreate their analysis [here]()



- Only a 107 people have downloaded this cool dataset

How did they do this exactly?

- The linear method is the logistic regression (LR) [39] with Elastic Net regularisation.
- The first non-linear method is Support Vector regression (SVM) with a Radial Basis Function (RBF) kernel.
- Although a standard non-linear method used in regression, SVMs do not inform which features are the most important in our predictive task. For this reason, we use Gaussian Processes (GP) for regression. GPs formulate a Bayesian non-parametric statistical framework which defines a prior on functions. The properties of the functions are given by a kernel which models the covariance in the response values as a function of its inputs. In order to enable feature interpretability, we use the Squared Exponential (a. k.a. RBF) covariance function with Automatic Relevance Determination (ARD) to learn a separate kernel lengthscale for each feature. Intuitively, the lengthscale parameter controls the variation along that dimension, i.e. a low value makes the output very sen

Another example--

- Foster, D., and Mark Liberman Robert A. Stine. "Featurizing Text: Converting Text into Predictors for Regression Analysis." *Wharton School of the University of Pennsylvania* (2013).

You have data like this (n=7500):

```
$399000 Stunning skyline views like something from a postcard are yours
with this large 2 bed, 2 bath loft in Dearborn Tower!  Detailed
hrdwd floors throughout the unit compliment an open kitchen and
spacious living-room and dining-room /w walk-in closet, steam
shower and marble entry.  Parking available.

$13000 4 bedroom, 2 bath 2 story frame home. Property features a
large kitchen, living-room and a full basement. This is a Fannie Mae
Homepath property.

$65000 Great short sale opportunity...  Brick 2 flat with 3 bdrm
each unit. 4 or more cars parking. Easy to show.

$29900 This 3 flat with all 3 bed units is truly a great
investment!! This property also comes with a full attic that has
the potential of a build-out-thats a possible 4 unit building in a
```

Tokenize the data

Go from this:

```
Brick flat, 2 bdrm.  With two-car garage.
```

Separated into tokens, this text becomes a list of 10 tokens representing 9 word types:

```
{brick, flat, <,>, 2, bdrm, <.>, with, two-car, garage,<.>}
```

Once tokenized, all characters are lower case. Punctuation symbols, such as commas and periods, are “words” in this sense. We leave embedded hyphens in place. Since little is known about rare words that are observed in only one or two documents, we represent their occurrence by the symbol ‘<UNK>’. The end of each document is marked by a unique type. We make no attempt to correct spelling errors and typos nor

Can you predict prices from words?

Table 2: *Multiple regression of log prices on counts from the document/word matrix W for the most common 2,000 words.* The table shows the 14 estimates that exceed the Bonferroni threshold for statistical significance.

	Estimate	Std. Error	t	$\Pr(>\|t\|)$
vacant	-0.5518	0.0652	-8.46	0.0000
deed	-1.3155	0.1557	-8.45	0.0000
OOV	0.0373	0.0059	6.33	0.0000
units	0.1929	0.0342	5.64	0.0000
discount	-1.4959	0.2992	-5.00	0.0000
investment	-0.2334	0.0497	-4.70	0.0000
most	0.3350	0.0736	4.55	0.0000
bucktown	0.3570	0.0790	4.52	0.0000
sf	0.3305	0.0741	4.46	0.0000
pullman	-0.6244	0.1423	-4.39	0.0000
bedroom	-0.0978	0.0227	-4.31	0.0000

$s_e = 0.682$ with $R^2 = 0.766$, $\overline{R}^2 = 0.681$

Then they extend this

1- Compute matrices that (a) count the number of times that word types appear within each document (such as a real estate listing) and (b) count the number of times that word types are found adjacent to each other.

2- Compute truncated singular value decompositions (SVD) of the resulting matrices of counts. The leading singular vectors of these decompositions are our regressors.

Two considerations

Once the source text has been tokenized, we form two matrices of counts. The SVD of each of these defines a set of explanatory variables. The matrices, W and B, differ in how they measure the similarity of words. Words are judged to be similar if they appear in the same context. For the document/word matrix W, the context is a document – a real estate listing. This matrix holds counts of which words appear in the same document, ignoring the order in which the words appear. This approach treats each document (or listing) as a bag of words, a multiset that does not distinguish the placement of the words. The second matrix adopts a very different perspective that relies entirely upon ordering; it defines the context by adjacency. The bigram matrix B counts how often words appear adjacent to each other. The document/word and bigram matrices thus represent two extremes of a common approach: Associate words that co-occur within some context. W uses the wide window provided by a document, whereas B uses the most narrow window possible. The wider window afforded by a document hints that W emphasizes semantic similarity, whereas the narrow window of adjacency that defines B suggests more emphasis on local syntax. Curiously, we find either approach effective and make use of both.

One more regression example

Rao, Delip, et al. "Classifying latent user attributes in twitter." *Proceedings of the 2nd international workshop on Search and mining user-generated contents*. ACM, 2010.

The question

- Can we predict gender from tweets?

- Note that this relies on a simple binary characterization of gender. There is a great need to do more in this regard to push beyond simplifications.

Yes-- ~70% accuracy vs. baseline 50%

- They adopt a sociolinguistic approach:

FEATURE	*Description/Example*
SIMLEYS	A list of emoticons compiled from the Wikipedia.
OMG	Abbreviation for 'Oh My God'
ELLIPSES	'....'
POSSESIVE BIGRAMS	E.g. my_XXX, our_XXX
REPATED ALPHABETS	E.g. niceeeeee, noooo waaaay
SELF	E.g., I_xxx, Im_xxx
LAUGH	E.g. LOL, ROTFL, LMFAO, haha, hehe
SHOUT	Text in ALLCAPS
EXASPERATION	E.g. Ugh, mmmm, hmmm, ahh, grrr
AGREEMENT	E.g. yea, yeah, ohya
HONORIFICS	E.g. dude, man, bro, sir
AFFECTION	E.g. xoxo
EXCITEMENT	A string of exclamation symbols (!!!!!)
SINGLE EXCLAIM	A single exclamation at the end of the tweet
PUZZLED PUNCT	A combination of any number of ? and ! (!?!!??!)

Some differences

Feature	*#female/#male*
Emoticons	3.5
Elipses	1.5
Character repetition	1.4
Repeated exclamation	2.0
Puzzled punctuation	1.8
OMG	4.0

Some more differences

Disfluency/Agreement	*#female/#male*
oh	2.3
ah	2.1
hmm	1.6
ugh	1.6
grrr	1.3
yeah, yea, ...	0.8

A couple more papers to look at ...

1. D'Orazio, Vito, et al. "Separating the Wheat from the Chaff: Applications of Automated Document Classification Using Support Vector Machines." *Political Analysis* 22.2 (2014): 224242.
2. Monroe, Burt, Michael Colaresi, and Kevin Quinn. 2008. "Fightin' Words: Lexical Feature Selection and Evaluation for Identifying the Content of Political Conflict". *Political Analysis*16(4))

3. Word to vector (Word2vec) models

Word2vec models combine many things that we have discussed

- The Wharton group's paper a few minute ago already starts to point in this direction…

- … as did our discussion of topic models

Word embeddings look to maintain high level semantic connectivity

- I am borrowing from Chris Bail on this

- In their most basic form, word embeddings are a technique for identifying similarities between words in a corpus by using some type of model to predict the co-occurence of words within a small chunk of text.

- There are a few different ways to get at words embeddings, using different “context windows” and different algorithms that try to predict the context words given the center word

Word embeddings look to maintain high level semantic connectivity

- For a great example, check out here:

- We take hundreds of thousands of words and we reduce them down to a couple dozen or maybe a bit more dimensions that capture *most* of the most critical connections among the words - this can be done through singular value decomposition or factorization, which is similar in spirit to PCA

Word embeddings look to maintain high level semantic connectivity

What words are closest to `"error"` in the data set of CFPB complaints, as determined by our word embeddings?

```
tidy_word_vectors %>%
  nearest_neighbors("error")
```

```
#> # A tibble: 7,475 × 2
#>    item1             value
#>    <chr>             <dbl>
#>  1 error             1
#>  2 mistake           0.683
#>  3 clerical          0.627
#>  4 problem           0.582
#>  5 glitch            0.580
#>  6 errors            0.571
#>  7 miscommunication  0.512
#>  8 misunderstanding  0.486
#>  9 issue             0.478
#> 10 discrepancy       0.474
#> # … with 7,465 more rows
```

Mistakes, problems, glitches – sounds bad!

- What we can do, is look for most strongly associated words with some focal word, like “error”

A social science example

- For a great example, check out [here](#):

PNAS

ARTICLES FRONT MATTER AUTHORS TOPICS SIGN IN SUBMIT

RESEARCH ARTICLE | SOCIAL SCIENCES | FREE ACCESS

Word embeddings quantify 100 years of gender and ethnic stereotypes

Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou Authors Info & Affiliations

April 3, 2018 | 115 (16) | https://doi.org/10.1073/pnas.1720347115

Significance

Word embeddings are a popular machine-learning method that represents each English word by a vector, such that the geometry between these vectors captures semantic relations between the corresponding words. We demonstrate that word embeddings can be used as a powerful tool to quantify historical trends and social change. As specific applications, we develop metrics based on word embeddings to characterize how gender

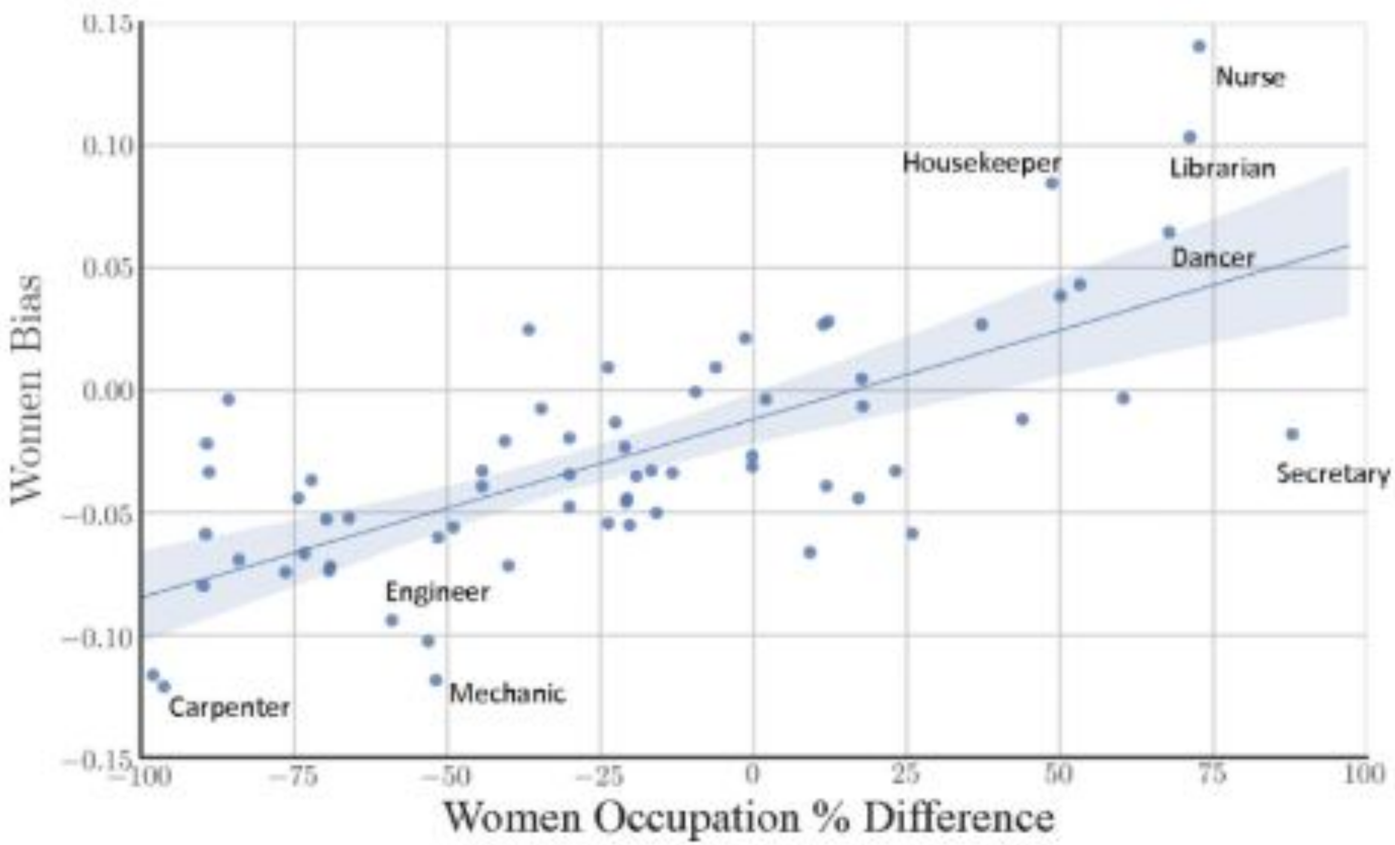
PNAS

Vol. 115 | No. 16

Significance

Occupations are “coded” as female

Fig. 1.



Women's occupation relative percentage vs. embedding bias in Google News vectors. More positive indicates more associated with women on both axes. $P < 10^{-10}$, $r^2 = 0.499$. *The shaded region is the* 95% *bootstrapped confidence interval of the regression line. In this single embedding, then, the association in the embedding effectively captures the percentage of women in an occupation.*

There is much more to word embeddings as well

- There are many other potential elements to building word embedding models too

- This is just a taste of what is possible

Agenda

1. How do I get text again?
2. Running regressions on all these words
3. What is natural language processing (NLP) vs. text analysis?

BTW, how about this?

Polygraph

BY MATT DANIELS

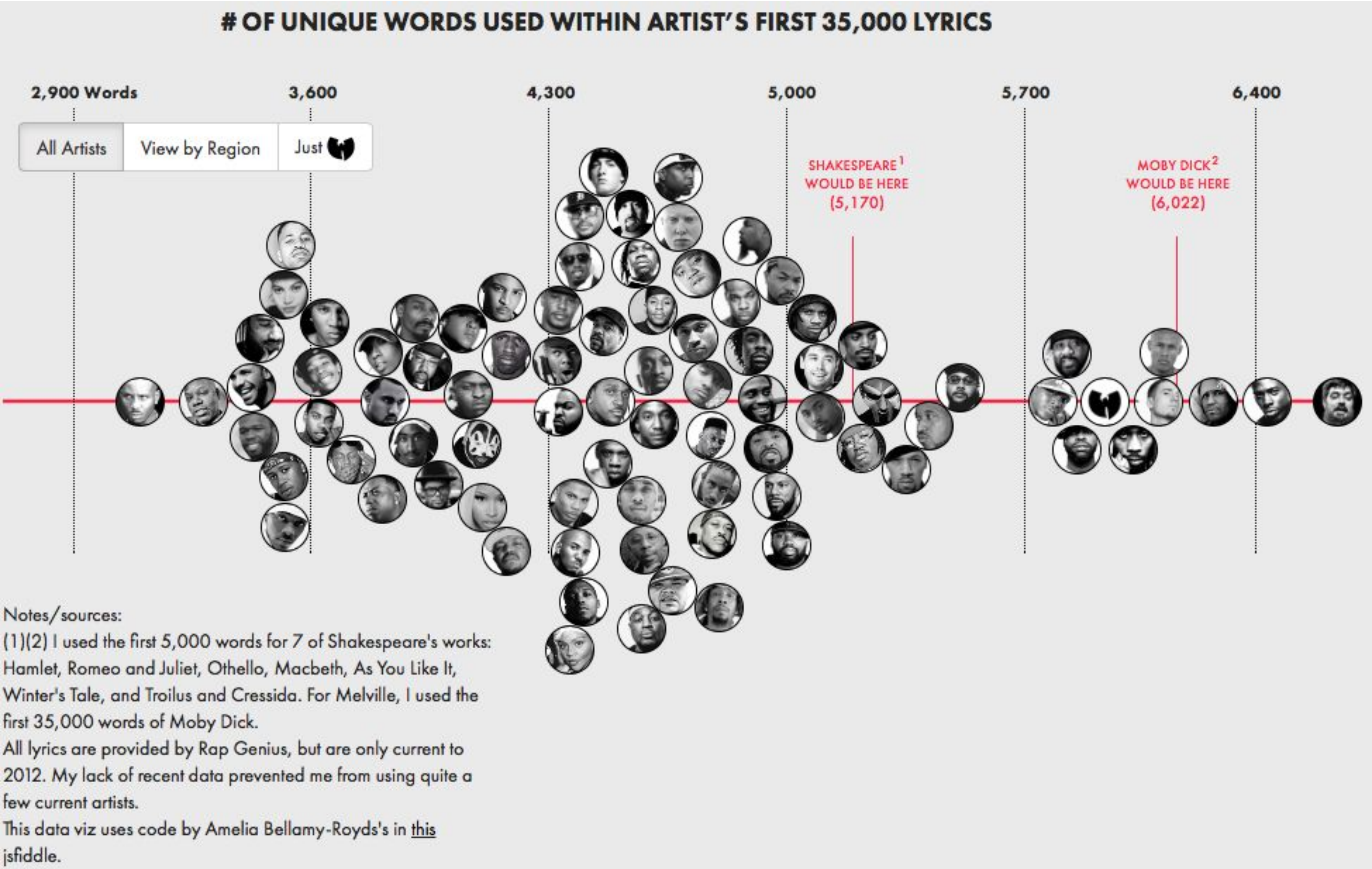
THE LARGEST VOCABULARY

IN HIP HOP

RAPPERS, RANKED BY THE NUMBER OF UNIQUE WORDS USED IN THEIR LYRICS

Literary elites love to rep Shakespeare's vocabulary: across his entire corpus, he uses 28,829 words, suggesting he knew over 100,000 words and arguably had the largest vocabulary, ever.

I decided to compare this data point against the most famous artists in hip hop. I used each artist's first 35,000 lyrics. That way, prolific artists, such as Jay-Z, could be compared to newer artists, such as Drake.



OF UNIQUE WORDS USED WITHIN ARTIST'S FIRST 35,000 LYRICS
2,900 Words
3,600
4,300
5,000
5,700
6,400
All Artists
View by Region
Just
SHAKESPEARE[1] WOULD BE HERE (5,170)
MOBY DICK[2] WOULD BE HERE (6,022)
Notes/sources:
(1)(2) I used the first 5,000 words for 7 of Shakespeare's works: Hamlet, Romeo and Juliet, Othello, Macbeth, As You Like It, Winter's Tale, and Troilus and Cressida. For Melville, I used the first 35,000 words of Moby Dick.
All lyrics are provided by Rap Genius, but are only current to 2012. My lack of recent data prevented me from using quite a few current artists.
This data viz uses code by Amelia Bellamy-Royds's in this jsfiddle.

#1 - AESOP ROCK

4. Why text analysis again?

5. The specter of LLMs looms over all of this