Irene Alvarado
CMU HCI

# Visualization in HCI Final Project

## Visualizing 14 Years of Venezuelan ex-president Hugo Chavez's Rule

# Overview

With my proposed visualization, I aim to show a textual analysis of 14 years of presidential rule of Hugo Chavez in Venezuela. Showcased as a website, users will be able to see how his use of language changed over the years, as well as topics that became important in different moments. The visualization is meant to be exploratory and a starting point for Venezuelan researchers.

The final visualization can be found here:
https://irealva.github.io/hugo-chavez-speeches-analysis/web/index3.html

The github repo is here:
https://github.com/irealva/hugo-chavez-speeches-analysis

# Motivation

Hugo Chavez ruled as Venezuela’s president from 1999 until his death in 2013. A controversial figure, he was both hated and adored by Venezuelans who saw him gain power over the years and institute his vision of a socialist revolution. His personality and political ideas were perhaps best appreciated in his speeches - many of which lasted several hours and were televised as a “cadena”, literally a “chain” in english and meaning that all news channels had to stream the speeches. His longest speech: 9 hours and 30 minutes. The chavista government left behind a trove of data on the thoughts, motivations, and rhetoric of its leader.

The problem is that many of these speeches were dutifully transcribed, but often not visualized or analyzed. They remain locked in PDFs and government websites, and in Spanish. My project aims to visualize how Chavez’s language changed over time in his nearly 14-year rule. As an initial step, the visualization is meant to be exploratory and to help any given Venezuelan or researcher with a basic knowledge of the country to be able to search for trends and insights. In so doing, I also aim to create a dataset that other researchers (especially english-speaking ones) might be able to use.

# Questions to Answer

At the outset I was wondering questions such as:

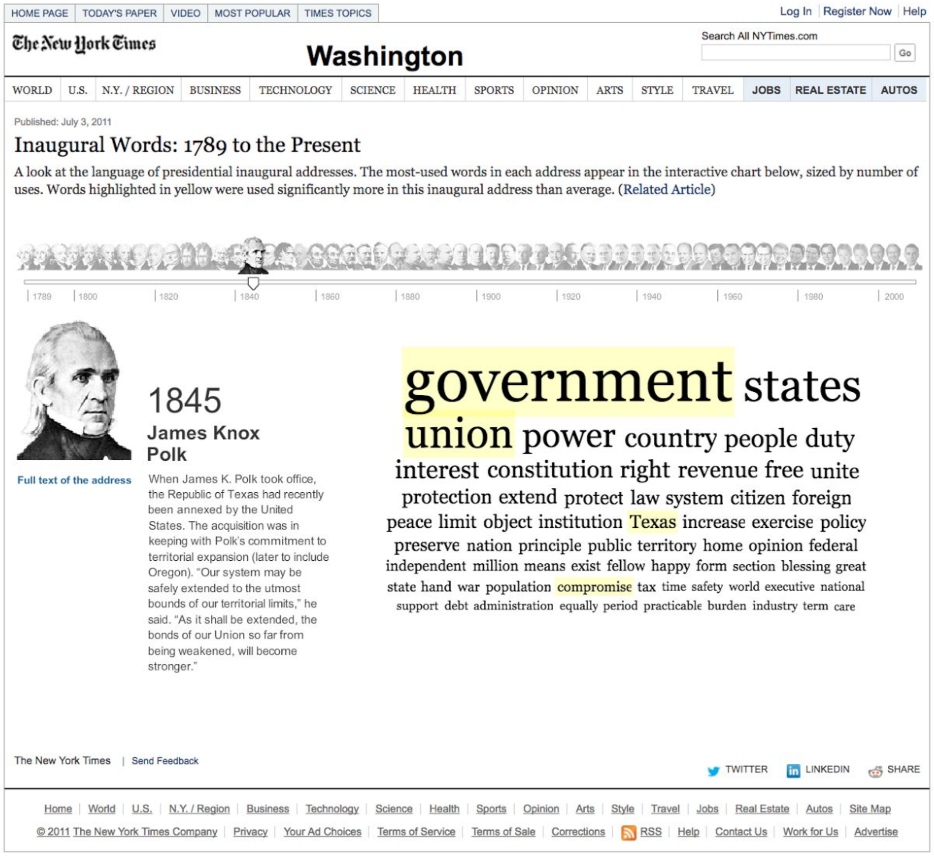
- How often does Chavez speak of socialism and communism? How early in his career did the socialist rhetoric begin?
- Given that his government became more and more confrontational with the opposition - many of whose leaders were placed in jail as political prisoners - I wanted to know whether his language became more violent, confrontational, warlike over time. Phrases like “patria, socialismo o muerte” / “the homeland, socialism, or death” slowly made their way into the Venezuelan consciousness.
- How often did he refer to the ills that plagued Venezuela in the later years of his rule? Crime, inflation, scarcity, the weakening economy…
- How can I get a sense of how long his speeches were and how that might have changed over time? Around what time period did he engage in his longest speeches?

As I started digging into the data and gaining some preliminary insights from data analysis, I started adding more questions to my initial list:

- Do Chavez’s international speeches (say to the U.N. or in the middle east) differ greatly in language from his national ones?
- Is he more or less formal in his use of language when giving an “official” statewide speech versus his more informal T.V. show “Alo Presidente”? As a note, Chavez was actually featured weekly on a T.V. show called “Hello Mr. President”. According to some estimates thanks to the show Chavez spent an average of 40 hours per week on television.
- What types of writers, thinkers, historians, politicians did he most often refer to?
- When did the U.S. become Venezuela’s number #1 enemy according to Chavez? At some point in his presidency, Chavez started blaming all of Venezuela’s economic woes on American imperialism.

# Related Work

As usual, the New York Times provided excellent inspiration for where to start. In “Inaugural Words” the Times visualizes about 200 years worth of presidential inaugural speeches. Most-used words are shown in a type of descending word cloud.



Published: July 3, 2011

Inaugural Words: 1789 to the Present

A look at the language of presidential inaugural addresses. The most-used words in each address appear in the interactive chart below, sized by number of uses. Words highlighted in yellow were used significantly more in this inaugural address than average. (Related Article)

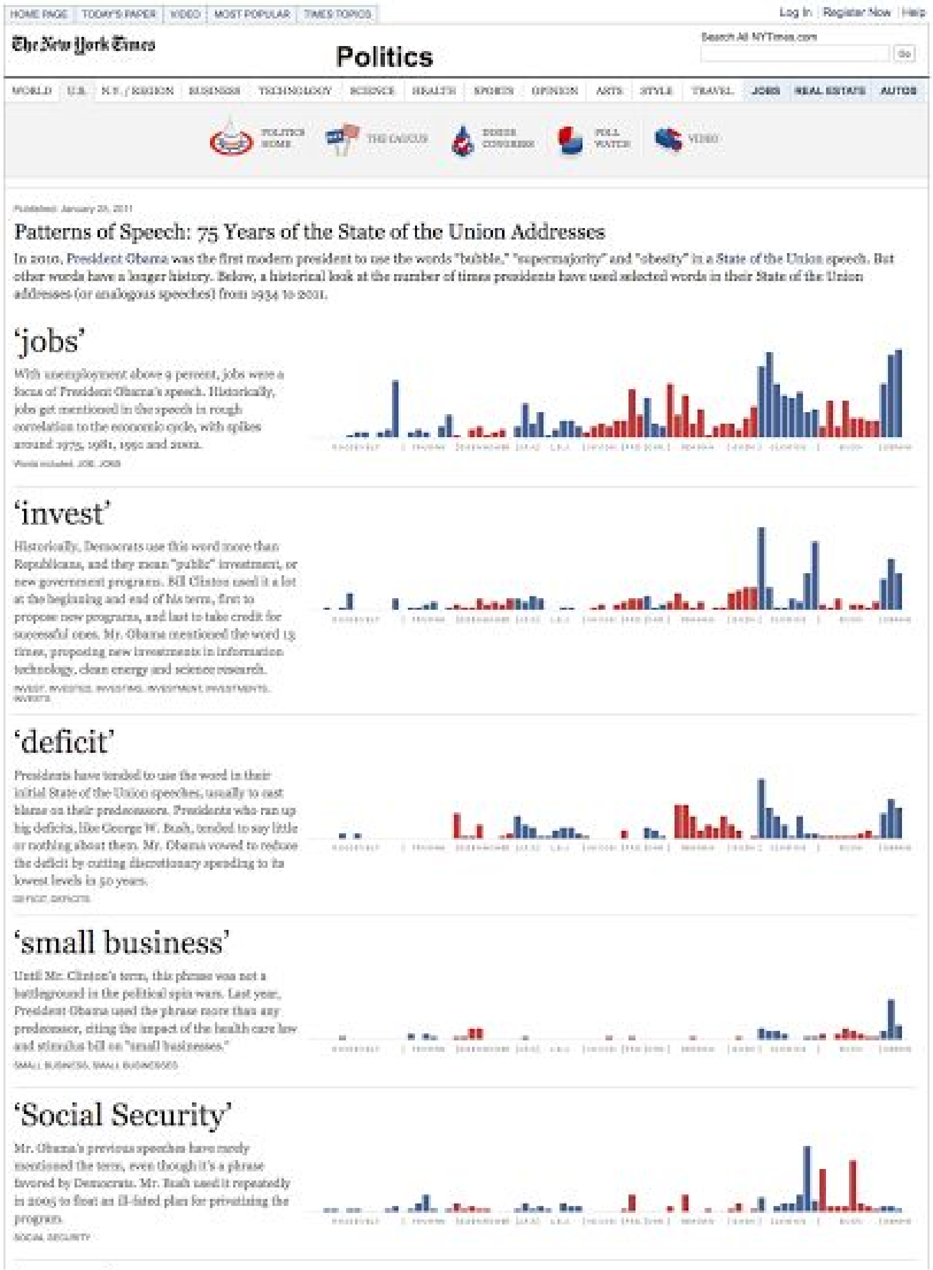
1845

James Knox Polk

When James K. Polk took office, the Republic of Texas had recently been annexed by the United States. The acquisition was in keeping with Polk's commitment to territorial expansion (later to include Oregon). “Our system may be safely extended to the utmost bounds of our territorial limits,” he said. “As it shall be extended, the bonds of our Union so far from being weakened, will become stronger.”

From: http://www.nytimes.com/interactive/2009/01/17/washington/20090117_ADDRESSES.html?_r=0

Another key inspiration was the “Patterns of Speech” visualization. Here, the Times selected key words in State of the Union addresses and plotted how those words' frequencies changed over time.



Published: January 25, 2011

Patterns of Speech: 75 Years of the State of the Union Addresses

In 2010, President Obama was the first modern president to use the words "bubble," "supermajority" and "obesity" in a State of the Union speech. But other words have a longer history. Below, a historical look at the number of times presidents have used selected words in their State of the Union addresses (or analogous speeches) from 1934 to 2011.

'jobs'

With unemployment above 9 percent, jobs were a focus of President Obama's speech. Historically, jobs get mentioned in the speech in rough correlation to the economic cycle, with spikes around 1975, 1981, 1992 and 2002.

Words included: JOB, JOBS

'invest'

Historically, Democrats use this word more than Republicans, and they mean "public" investment, or new government programs. Bill Clinton used it a lot at the beginning and end of his term, first to propose new programs, and last to take credit for successful ones. Mr. Obama mentioned the word 13 times, proposing new investments in information technology, clean energy and science research.

INVEST, INVESTED, INVESTING, INVESTMENT, INVESTMENTS, INVESTS

'deficit'

Presidents have tended to use the word in their initial State of the Union speeches, usually to cast blame on their predecessors. Presidents who ran up big deficits, like George W. Bush, tended to say little or nothing about them. Mr. Obama vowed to reduce the deficit by cutting discretionary spending to its lowest levels in 50 years.

DEFICIT, DEFICITS

'small business'

Until Mr. Clinton's term, this phrase was not a battleground in the political spin wars. Last year, President Obama used the phrase more than any predecessor, citing the impact of the health care law and stimulus bill on "small businesses."

SMALL BUSINESS, SMALL BUSINESSES

'Social Security'

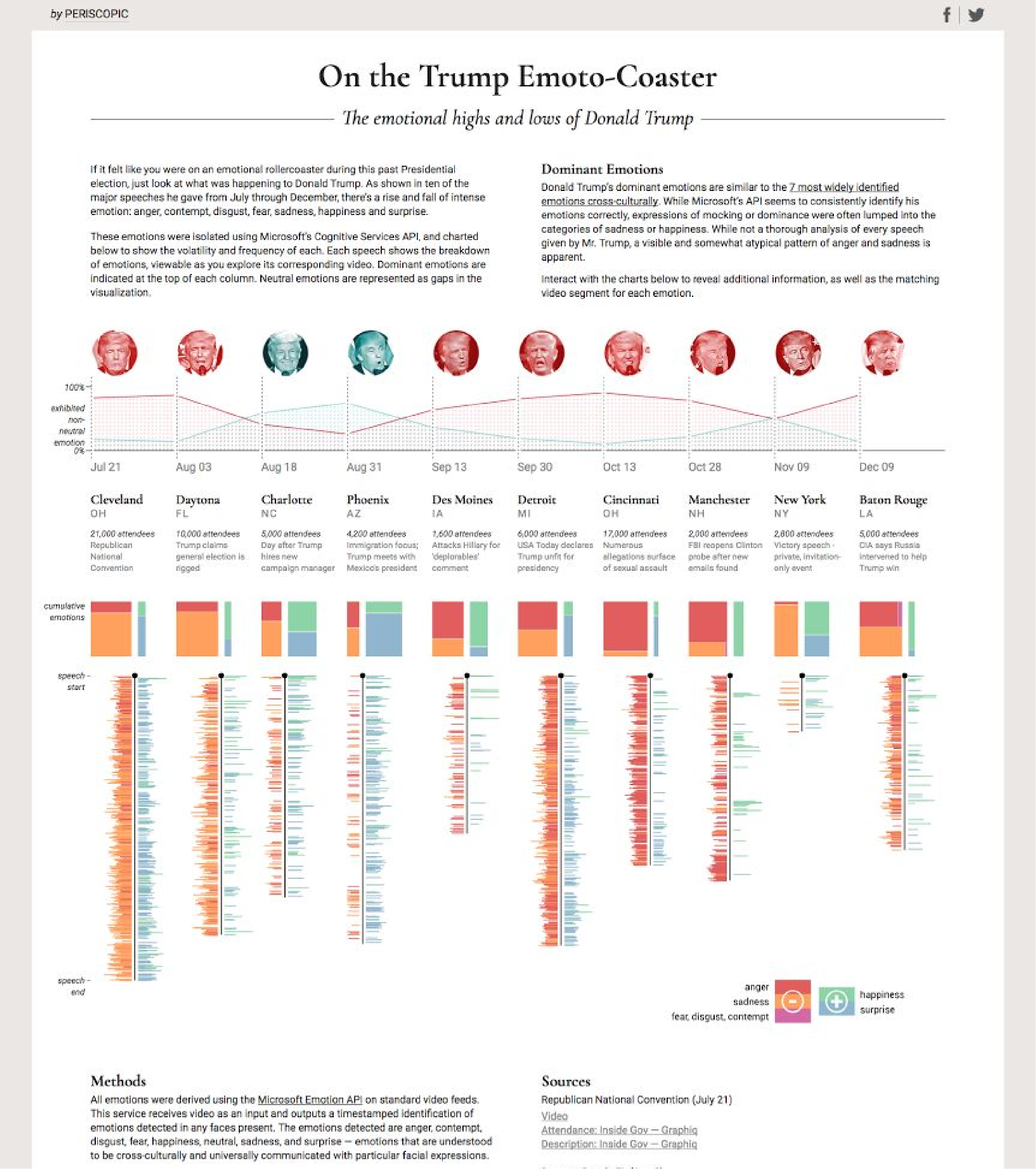
Mr. Obama's previous speeches have rarely mentioned the term, even though it's a phrase favored by Democrats. Mr. Bush used it repeatedly in 2005 to float an ill-fated plan for privatizing the program.

SOCIAL SECURITY

From: http://www.nytimes.com/interactive/2011/01/25/us/politics/state-of-the-union-words-used.html

Here, I started to think of ways I could combine both visualizations. What struck me was how simple they were - but in that simplicity the data exposed was extremely clear. The issue for me was that each visualization was meant to be a small snapshot into the data while I was thinking of a more holistic visualization that might allow you to look at the same data in multiple ways.

Finally, I found a slightly more complex visualization of Trump speeches in "On the Trump Emoto-Coaster". It visualizes the emotional quality of Trump's speeches, by essentially plotting what percentage of his speeches could be characterized with emotions such as: anger, surprise, sadness, happiness.



by PERISCOPIC

# On the Trump Emoto-Coaster

*The emotional highs and lows of Donald Trump*

If it felt like you were on an emotional rollercoaster during this past Presidential election, just look at what was happening to Donald Trump. As shown in ten of the major speeches he gave from July through December, there's a rise and fall of intense emotion: anger, contempt, disgust, fear, sadness, happiness and surprise.

These emotions were isolated using Microsoft's Cognitive Services API, and charted below to show the volatility and frequency of each. Each speech shows the breakdown of emotions, viewable as you explore its corresponding video. Dominant emotions are indicated at the top of each column. Neutral emotions are represented as gaps in the visualization.

## Dominant Emotions

Donald Trump's dominant emotions are similar to the 7 most widely identified emotions cross-culturally. While Microsoft's API seems to consistently identify his emotions correctly, expressions of mocking or dominance were often lumped into the categories of sadness or happiness. While not a thorough analysis of every speech given by Mr. Trump, a visible and somewhat atypical pattern of anger and sadness is apparent.

Interact with the charts below to reveal additional information, as well as the matching video segment for each emotion.

| Jul 21 | Aug 03 | Aug 18 | Aug 31 | Sep 13 | Sep 30 | Oct 13 | Oct 28 | Nov 09 | Dec 09 |
|---|---|---|---|---|---|---|---|---|---|
| Cleveland<br>OH | Daytona<br>FL | Charlotte<br>NC | Phoenix<br>AZ | Des Moines<br>IA | Detroit<br>MI | Cincinnati<br>OH | Manchester<br>NH | New York<br>NY | Baton Rouge<br>LA |
| *21,000 attendees*<br>Republican National Convention | *10,000 attendees*<br>Trump claims general election is rigged | *5,000 attendees*<br>Day after Trump hires new campaign manager | *4,200 attendees*<br>Immigration focus; Trump meets with Mexico's president | *1,600 attendees*<br>Attacks Hillary for 'deplorables' comment | *6,000 attendees*<br>USA Today declares Trump unfit for presidency | *17,000 attendees*<br>Numerous allegations surface of sexual assault | *2,000 attendees*<br>FBI reopens Clinton probe after new emails found | *2,800 attendees*<br>Victory speech - private, invitation-only event | *5,000 attendees*<br>CIA says Russia intervened to help Trump win |

## Methods

All emotions were derived using the Microsoft Emotion API on standard video feeds. This service receives video as an input and outputs a timestamped identification of emotions detected in any faces present. The emotions detected are anger, contempt, disgust, fear, happiness, neutral, sadness, and surprise — emotions that are understood to be cross-culturally and universally communicated with particular facial expressions.

## Sources

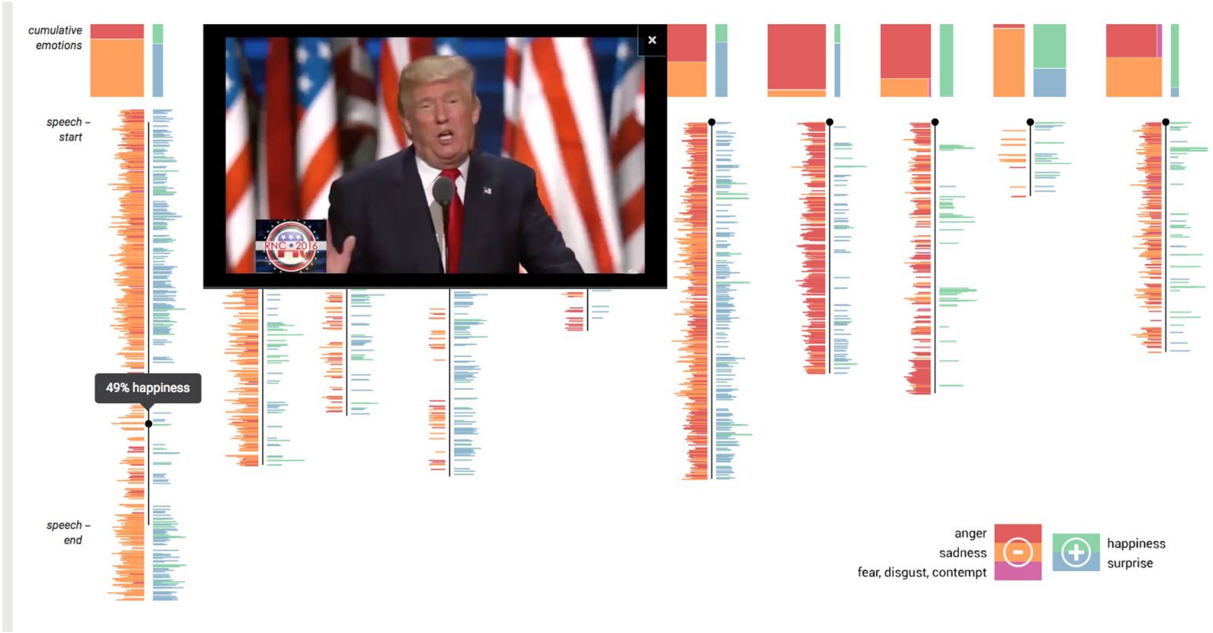
Republican National Convention (July 21)

Video

Attendance: Inside Gov — Graphiq

Description: Inside Gov — Graphiq

From: http://emotions.periscopic.com/



From: http://emotions.periscopic.com/

Here, I noticed that clicking on a specific part of a speech (visualized as a series of colored lines) would pull up a youtube video of that moment. I kept wondering about ways to provide a similar functionality in which a user might be able to see the big picture (an analysis based on a *collection* of speeches in a year) but dig deeper into individual speeches as well.

# Data

Obtaining and analyzing the data for this project was certainly a challenge and perhaps half the battle. I adequately felt the pain other data visualization professionals feel when declaring that so much of their time is spent on data wrangling.

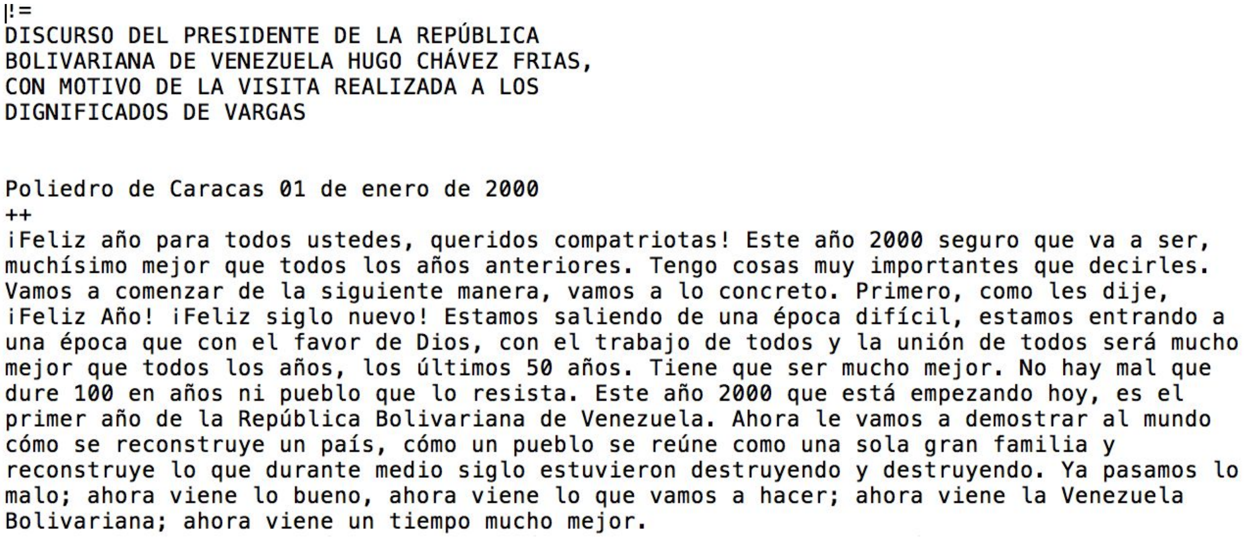
The data I had to obtain was mainly stored in two ways: a collection of PDFs or books published by the Venezuelan government containing Chavez's speeches and a government website listing individual speeches. I began by extracting information from PDFs mainly because I found that resource first. The limitation: I only had data from 1999-2006.

The covers of the PDFs I was using to parse the data

The PDF to text data process looked something like this:

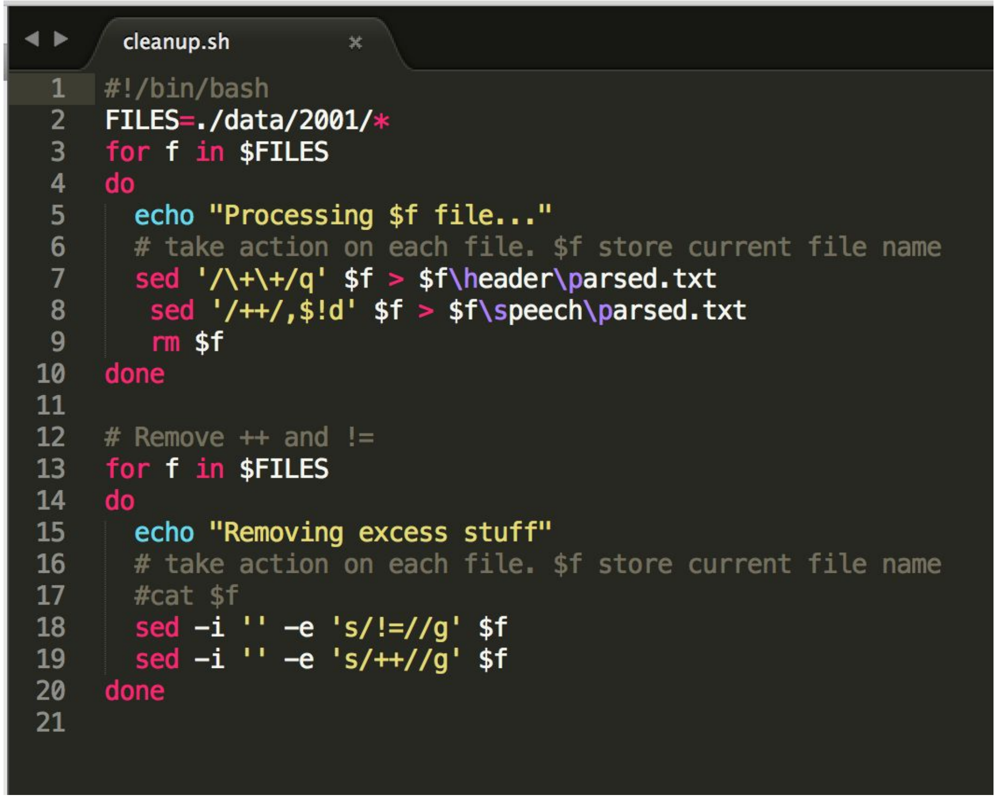
- Using Adobe Acrobat I saved the PDFs as text files.
- For some reason Acrobat was corrupting the encoding of the text, so I copy pasted the text onto Google Docs and downloaded the docs as text files. That seemed to fix the encoding.

- Now I had huge text files - one for each year - containing all the speech data for a given year.
- Using regular expressions, I added characters (!= and ++) that delimited speech headers (containing a date, the title of the speech) with the actual speech content:



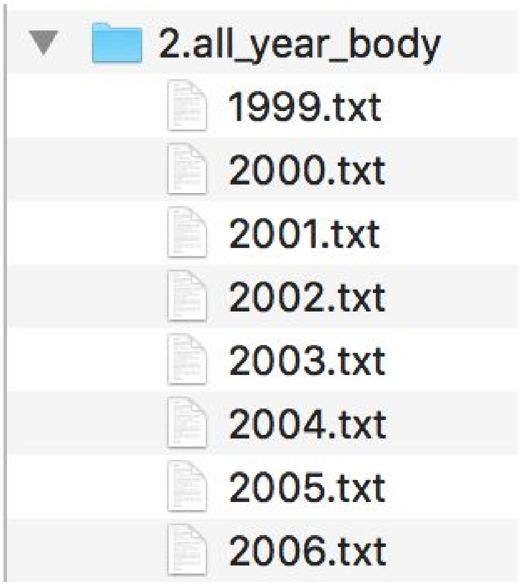
```
!=
DISCURSO DEL PRESIDENTE DE LA REPÚBLICA
BOLIVARIANA DE VENEZUELA HUGO CHÁVEZ FRIAS,
CON MOTIVO DE LA VISITA REALIZADA A LOS
DIGNIFICADOS DE VARGAS


Poliedro de Caracas 01 de enero de 2000
++
¡Feliz año para todos ustedes, queridos compatriotas! Este año 2000 seguro que va a ser,
muchísimo mejor que todos los años anteriores. Tengo cosas muy importantes que decirles.
Vamos a comenzar de la siguiente manera, vamos a lo concreto. Primero, como les dije,
¡Feliz Año! ¡Feliz siglo nuevo! Estamos saliendo de una época difícil, estamos entrando a
una época que con el favor de Dios, con el trabajo de todos y la unión de todos será mucho
mejor que todos los años, los últimos 50 años. Tiene que ser mucho mejor. No hay mal que
dure 100 en años ni pueblo que lo resista. Este año 2000 que está empezando hoy, es el
primer año de la República Bolivariana de Venezuela. Ahora le vamos a demostrar al mundo
cómo se reconstruye un país, cómo un pueblo se reúne como una sola gran familia y
reconstruye lo que durante medio siglo estuvieron destruyendo y destruyendo. Ya pasamos lo
malo; ahora viene lo bueno, ahora viene lo que vamos a hacer; ahora viene la Venezuela
Bolivariana; ahora viene un tiempo mucho mejor.
```

- Then I used a script to extract the headers out of the text files:



```
#!/bin/bash
FILES=./data/2001/*
for f in $FILES
do
  echo "Processing $f file..."
  # take action on each file. $f store current file name
  sed '/\+\+/q' $f > $f\header\parsed.txt
   sed '/++/,$!d' $f > $f\speech\parsed.txt
   rm $f
done

# Remove ++ and !=
for f in $FILES
do
  echo "Removing excess stuff"
  # take action on each file. $f store current file name
  #cat $f
  sed -i '' -e 's/!=//g' $f
  sed -i '' -e 's/++//g' $f
done

```

That’s how I obtained a list of all of Chavez’s speeches between the years of 1999-2006.



```
2.all_year_body
    1999.txt
    2000.txt
    2001.txt
    2002.txt
    2003.txt
    2004.txt
    2005.txt
    2006.txt
```

## Exploratory Data Analysis

While I had a obtained an initial dataset to begin my exploration, I spent quite some time performing basic textual analysis on the data: things like obtaining the most frequent words, finding bigrams, removing stopwords, and term frequency-inverse document frequency (tf-idf).

My tool of choice for this section was python and nltk, a natural language toolkit providing lots of great pre-built functions for text analysis.

### Cleanup

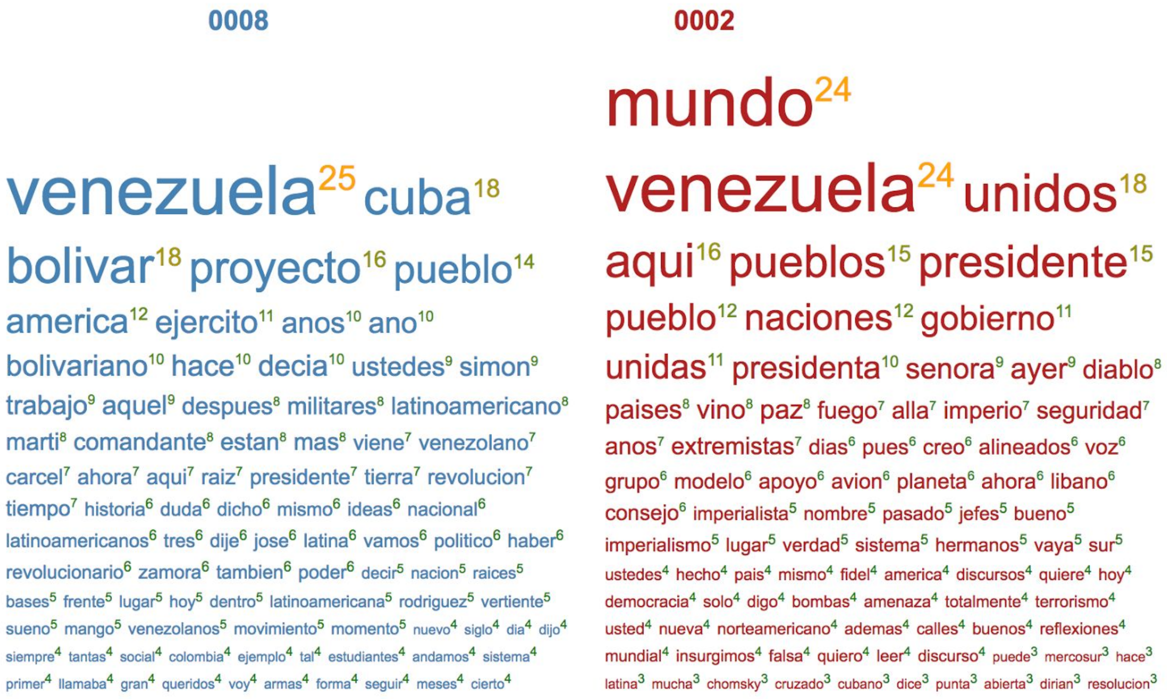
As an initial step to prepare for analysis, I made sure I removed all the accents from the files and converted all the words to lowercase. These steps were important in order to properly remove stopwords and count word frequency.

I chose to analyze the data in Spanish first, given that a lot of the meaning and sense of the words would have been lost in translation. I decided that the best use of translation would have to happen in the final visualization, i.e. on the webpage.

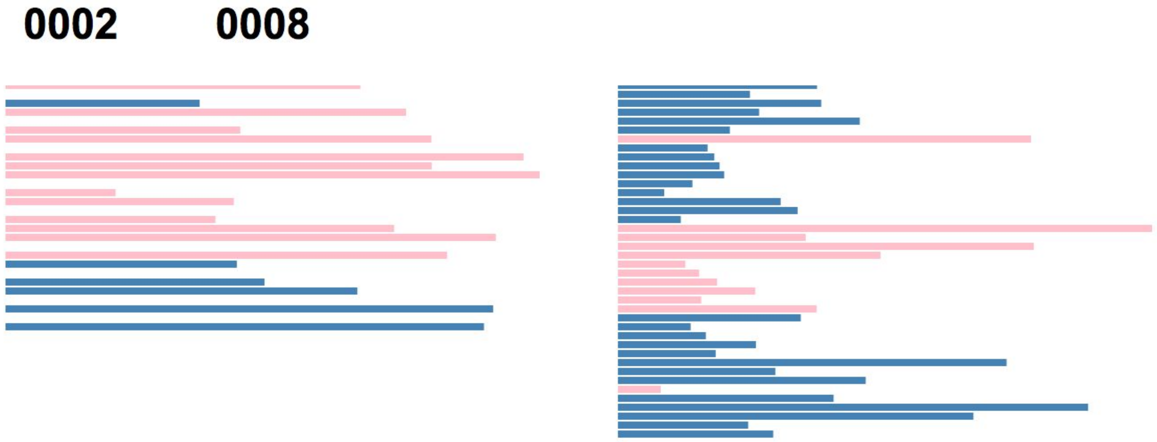
### Initial Exploration - Analyzing Individual Speeches

I first started exploring individual speeches and what kind of information I might extract from them.

This is an initial word list I created comparing two *individual* speeches in one single year:



This is another initial visualization comparing the length of two speeches. The text in pink is supposed to be dialogue, but the implementation for quote detection is somewhat crude and didn't always work correctly:



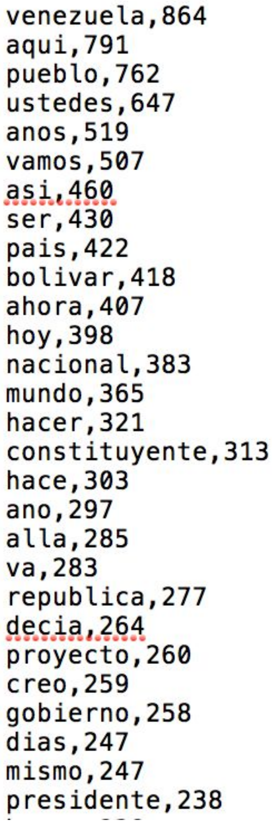
These initial explorations led me to realize that a *speech to speech comparison would be too overwhelming* given I had about 300 speeches just from 1999 to 2006. I shifted gears and started focusing on analyzing a *collection* of speeches organized by year.

## Exploring Collections of Speeches per Year

Most of my work can be found in the following iPython script: https://github.com/irealva/hugo-chavez-speeches-analysis/blob/master/python/yearly_speech_analysis.ipynb

*1. Most Commonly used Words:*

An initial analysis of most-commonly used words revealed something like the following for the year 1999:

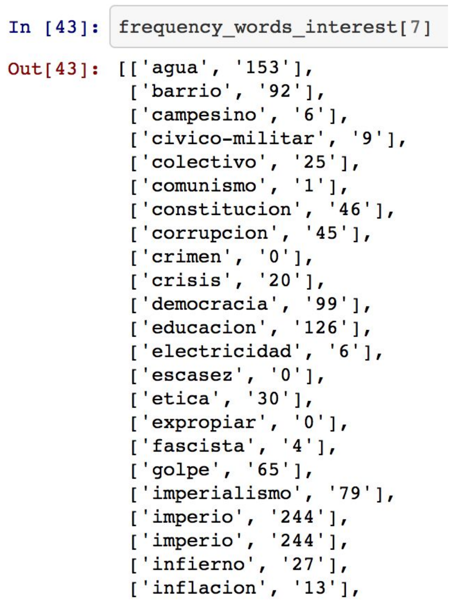


```
venezuela,864
aqui,791
pueblo,762
ustedes,647
anos,519
vamos,507
asi,460
ser,430
pais,422
bolivar,418
ahora,407
hoy,398
nacional,383
mundo,365
hacer,321
constituyente,313
hace,303
ano,297
alla,285
va,283
republica,277
decia,264
proyecto,260
creo,259
gobierno,258
dias,247
mismo,247
presidente,238
```

Another realization was that visualizing most commonly-used words would not be too interesting. The usual suspects appeared as most commonly-used words: “Venezuela”, a few stopwords that made it in like “here”, “years”, “the people”. By themselves they were not so revealing, which made me consider other options for textual analysis.

*2. Selected Words of Interest*

Instead of visualizing the most common words, I considered simply tracking the frequency of a selected list of words that I hand-picked. Using iPython notebooks, I could easily see the results of applying such a search to speeches in one given year:



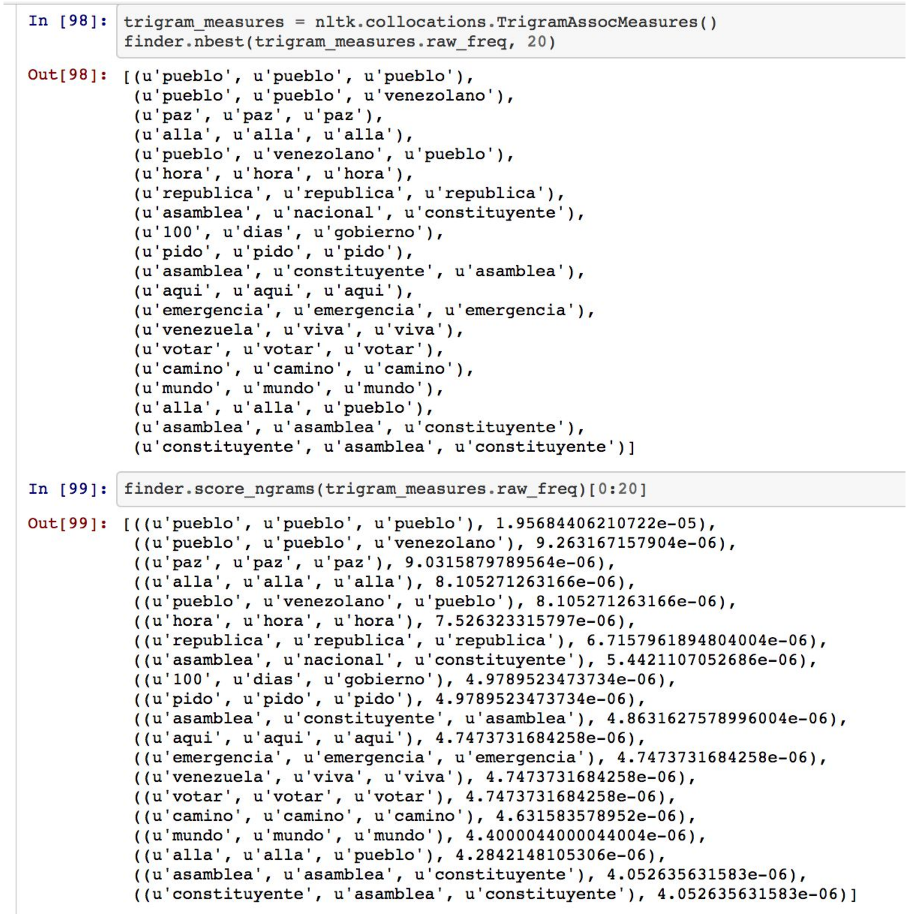
```
In [43]: frequency_words_interest[7]

Out[43]: [['agua', '153'],
          ['barrio', '92'],
          ['campesino', '6'],
          ['civico-militar', '9'],
          ['colectivo', '25'],
          ['comunismo', '1'],
          ['constitucion', '46'],
          ['corrupcion', '45'],
          ['crimen', '0'],
          ['crisis', '20'],
          ['democracia', '99'],
          ['educacion', '126'],
          ['electricidad', '6'],
          ['escasez', '0'],
          ['etica', '30'],
          ['expropiar', '0'],
          ['fascista', '4'],
          ['golpe', '65'],
          ['imperialismo', '79'],
          ['imperio', '244'],
          ['imperio', '244'],
          ['infierno', '27'],
          ['inflacion', '13'],
```

In this case we see that “agua” or “water” at the very top appears 153 times in 2006 as compared to “crimen” or “crime” which appears 0 times.

*3. Bigrams and Trigrams*

Next, I tried computing bigrams and trigrams using NLTK. The bigrams data was hard to interpret for isolated years, so I delayed my analysis until I could visualize all the years together on a single page.

On the other hand, the trigram data I created seemed like junk and I don’t know enough about text analysis to know what it means. Essentially, in many cases the same word was pulled out as a trigram: “people people people”.



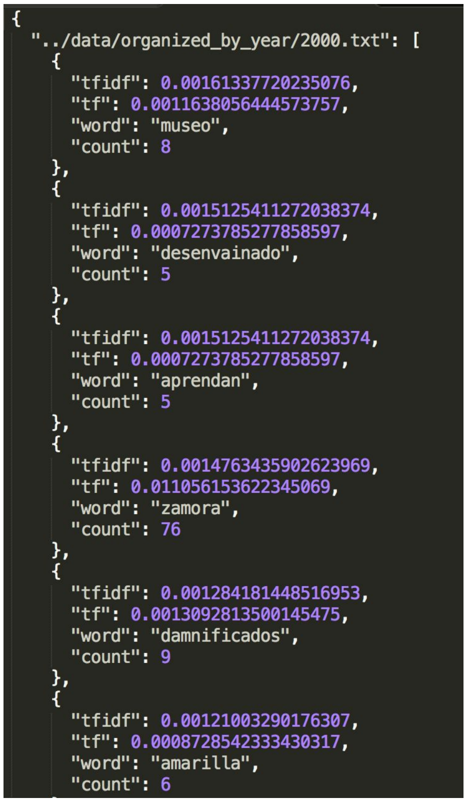
```
In [98]: trigram_measures = nltk.collocations.TrigramAssocMeasures()
         finder.nbest(trigram_measures.raw_freq, 20)

Out[98]: [(u'pueblo', u'pueblo', u'pueblo'),
          (u'pueblo', u'pueblo', u'venezolano'),
          (u'paz', u'paz', u'paz'),
          (u'alla', u'alla', u'alla'),
          (u'pueblo', u'venezolano', u'pueblo'),
          (u'hora', u'hora', u'hora'),
          (u'republica', u'republica', u'republica'),
          (u'asamblea', u'nacional', u'constituyente'),
          (u'100', u'dias', u'gobierno'),
          (u'pido', u'pido', u'pido'),
          (u'asamblea', u'constituyente', u'asamblea'),
          (u'aqui', u'aqui', u'aqui'),
          (u'emergencia', u'emergencia', u'emergencia'),
          (u'venezuela', u'viva', u'viva'),
          (u'votar', u'votar', u'votar'),
          (u'camino', u'camino', u'camino'),
          (u'mundo', u'mundo', u'mundo'),
          (u'alla', u'alla', u'pueblo'),
          (u'asamblea', u'asamblea', u'constituyente'),
          (u'constituyente', u'asamblea', u'constituyente')]

In [99]: finder.score_ngrams(trigram_measures.raw_freq)[0:20]

Out[99]: [((u'pueblo', u'pueblo', u'pueblo'), 1.95684406210722e-05),
          ((u'pueblo', u'pueblo', u'venezolano'), 9.263167157904e-06),
          ((u'paz', u'paz', u'paz'), 9.0315879789564e-06),
          ((u'alla', u'alla', u'alla'), 8.105271263166e-06),
          ((u'pueblo', u'venezolano', u'pueblo'), 8.105271263166e-06),
          ((u'hora', u'hora', u'hora'), 7.526323315797e-06),
          ((u'republica', u'republica', u'republica'), 6.7157961894804004e-06),
          ((u'asamblea', u'nacional', u'constituyente'), 5.4421107052686e-06),
          ((u'100', u'dias', u'gobierno'), 4.9789523473734e-06),
          ((u'pido', u'pido', u'pido'), 4.9789523473734e-06),
          ((u'asamblea', u'constituyente', u'asamblea'), 4.8631627578996004e-06),
          ((u'aqui', u'aqui', u'aqui'), 4.7473731684258e-06),
          ((u'emergencia', u'emergencia', u'emergencia'), 4.7473731684258e-06),
          ((u'venezuela', u'viva', u'viva'), 4.7473731684258e-06),
          ((u'votar', u'votar', u'votar'), 4.7473731684258e-06),
          ((u'camino', u'camino', u'camino'), 4.631583578952e-06),
          ((u'mundo', u'mundo', u'mundo'), 4.4000044000044004e-06),
          ((u'alla', u'alla', u'pueblo'), 4.2842148105306e-06),
          ((u'asamblea', u'asamblea', u'constituyente'), 4.052635631583e-06),
          ((u'constituyente', u'asamblea', u'constituyente'), 4.052635631583e-06)]
```

An example trigrams list from all the speeches in 1999

*4. Term Frequency-inverse Document Frequency*

I wanted to generate a list of the words that varied the most in each year of Chavez's rule. One way to do that is with a tf-idf analysis, which is one way to measure of the importance of a given word in a document that is part of a larger collection.

My tf-idf analysis took about a day to run and generated a huge json file that looks like the following, where "tfidf" is the measure of importance of a given word and "tf" is a measure of the term frequency:



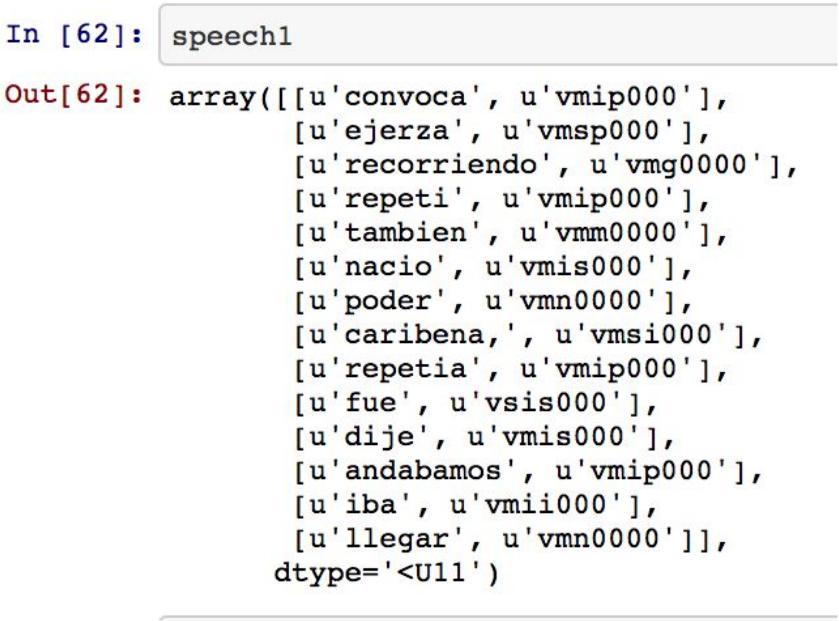
```
{
  "../data/organized_by_year/2000.txt": [
    {
      "tfidf": 0.00161337720235076,
      "tf": 0.0011638056444573757,
      "word": "museo",
      "count": 8
    },
    {
      "tfidf": 0.0015125411272038374,
      "tf": 0.0007273785277858597,
      "word": "desenvainado",
      "count": 5
    },
    {
      "tfidf": 0.0015125411272038374,
      "tf": 0.0007273785277858597,
      "word": "aprendan",
      "count": 5
    },
    {
      "tfidf": 0.0014763435902623969,
      "tf": 0.011056153622345069,
      "word": "zamora",
      "count": 76
    },
    {
      "tfidf": 0.001284181448516953,
      "tf": 0.0013092813500145475,
      "word": "damnificados",
      "count": 9
    },
    {
      "tfidf": 0.00121003290176307,
      "tf": 0.0008728542333430317,
      "word": "amarilla",
      "count": 6
```

I took the most important 30 words in each year (the highest tfidf) and decided to visualize them as a word cloud in the final visualization.

*5. Parts of Speech*

Finally, I used a Spanish model created by the Stanford Natural Language Processing Group to extract parts of speech (verbs, nouns, adjectives, etc.) from these speeches. My motivation here was to create word clouds of only verbs - thinking these might expose the kinds of actions and decisions Chavez was talking about each year.

The challenging aspect in this stage was dealing with how much time some of these analyses were taking. The data had to be parsed paragraph by paragraph and on my small laptop was taking whole days. I decided to leave the parts of speech analysis for the future.

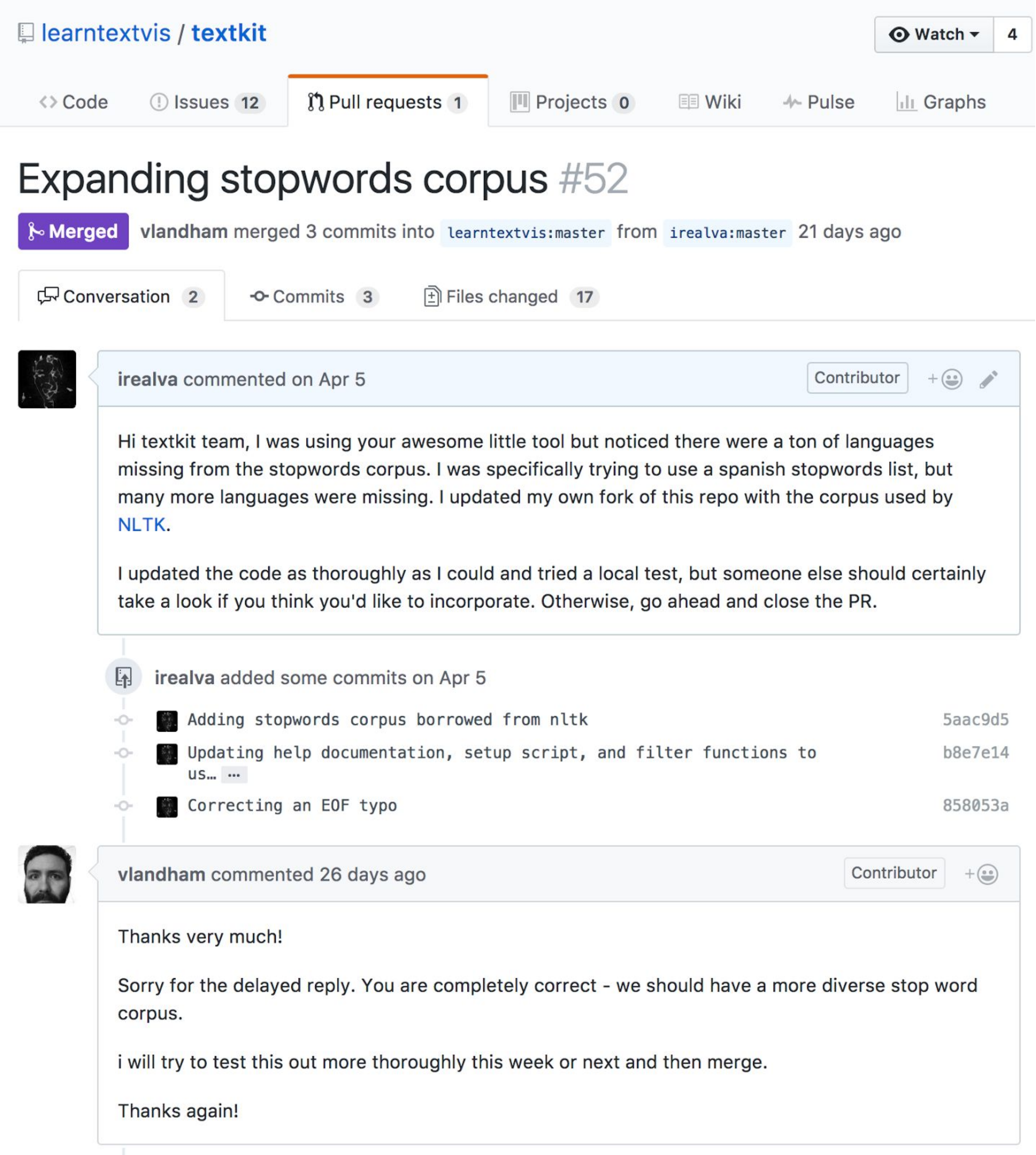


```
In [62]: speech1

Out[62]: array([[u'convoca', u'vmip000'],
                [u'ejerza', u'vmsp000'],
                [u'recorriendo', u'vmg0000'],
                [u'repeti', u'vmip000'],
                [u'tambien', u'vmm0000'],
                [u'nacio', u'vmis000'],
                [u'poder', u'vmn0000'],
                [u'caribena,', u'vmsi000'],
                [u'repetia', u'vmip000'],
                [u'fue', u'vsis000'],
                [u'dije', u'vmis000'],
                [u'andabamos', u'vmip000'],
                [u'iba', u'vmii000'],
                [u'llegar', u'vmn0000']],
               dtype='<U11')
```

All the verbs a sample paragraph in a speech in 1999

### Contributing to the community

As a side note, in my personal practice I've started thinking about ways I can give back to the open-source community that makes so many tools available for use. As part of my initial explorations I had been using a command line tool called *textvis* for parsing speech files. I noticed their stopwords corpus only included german and english, so I submitted a pull request to their repo with an expanded stopwords corpus as well as the right code edits to support more languages.



# Expanding stopwords corpus #52

Merged **vlandham** merged 3 commits into `learntextvis:master` from `irealva:master` 21 days ago

**irealva** commented on Apr 5

Hi textkit team, I was using your awesome little tool but noticed there were a ton of languages missing from the stopwords corpus. I was specifically trying to use a spanish stopwords list, but many more languages were missing. I updated my own fork of this repo with the corpus used by NLTK.

I updated the code as thoroughly as I could and tried a local test, but someone else should certainly take a look if you think you'd like to incorporate. Otherwise, go ahead and close the PR.

**irealva** added some commits on Apr 5

```
Adding stopwords corpus borrowed from nltk                          5aac9d5
Updating help documentation, setup script, and filter functions to  b8e7e14
us… …
Correcting an EOF typo                                              858053a
```

**vlandham** commented 26 days ago

Thanks very much!

Sorry for the delayed reply. You are completely correct - we should have a more diverse stop word corpus.

i will try to test this out more thoroughly this week or next and then merge.

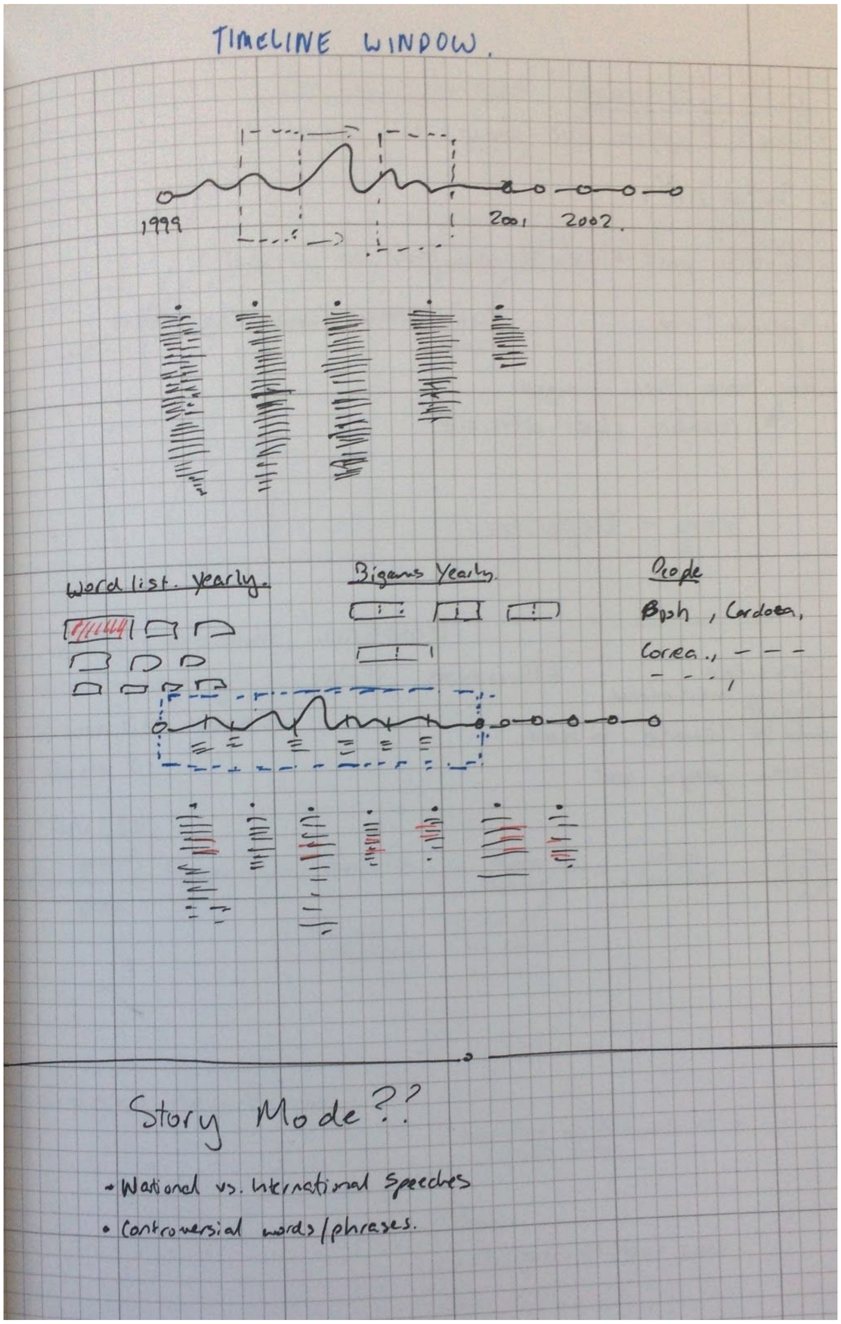
Thanks again!

# Design Evolution & Implementation

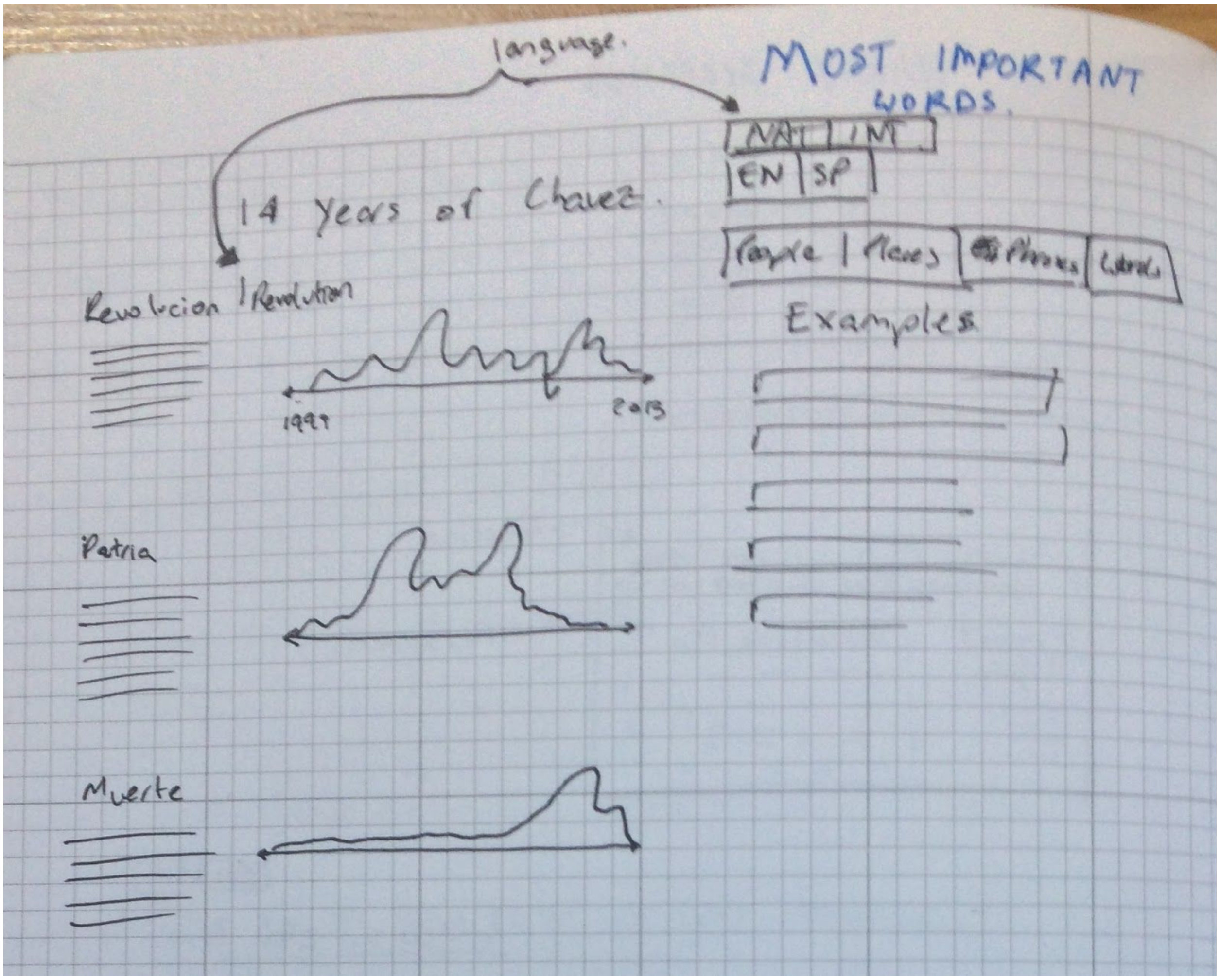
## Evolution

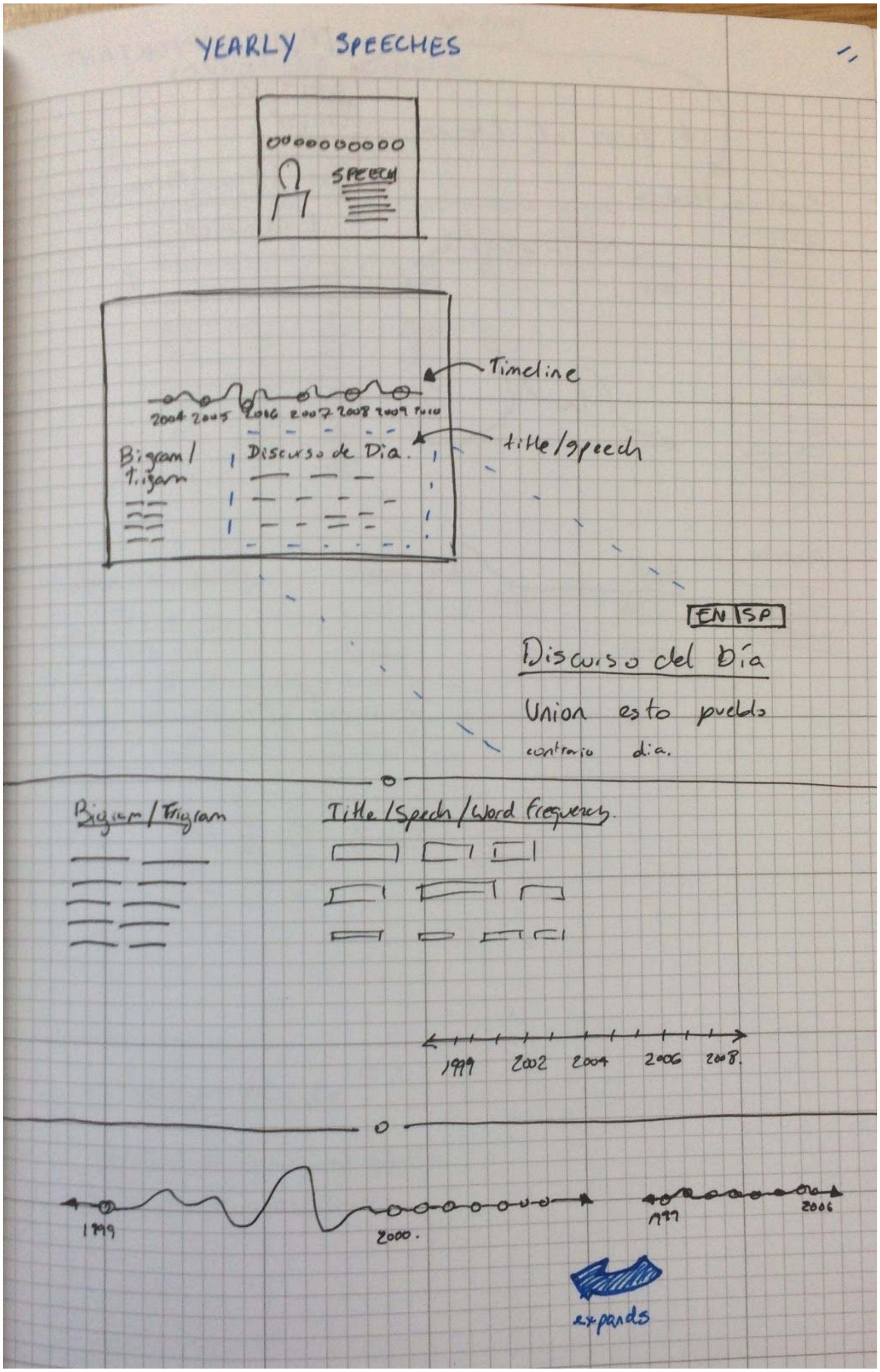
I started with some sketches of what I thought the final visualization could look like:

:

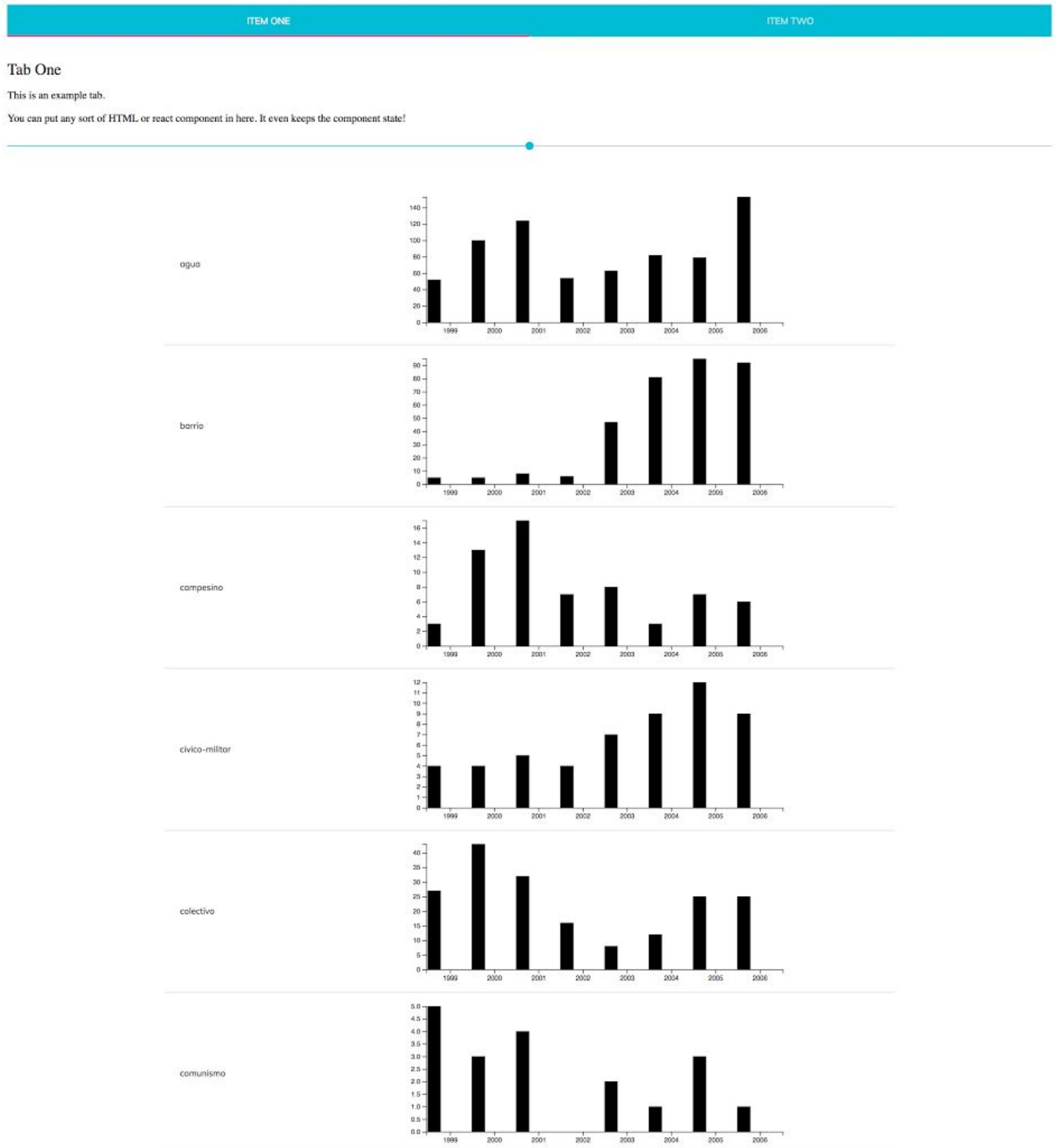


An initial idea involved a timeline view


language.
MOST IMPORTANT WORDS.
NAT | INT
EN | SP
People | Places
Phrases
Words
Examples
14 Years of Chavez.
Revolucion | Revolution
1999
2013
Patria
Muerte


YEARLY SPEECHES
SPEECH
Timeline
2004 2005 2006 2007 2008 2009 2010
title/speech
Bigram/ Trigam
Discurso de Dia.
EN SP
Discurso del Día
Union esto pueblo
contrario dia.
Bigram/Trigram
Title/Speech/Word Frequency.
1999 2002 2004 2006 2008.
1999
2000.
1999
2006
expands

Then I met with Professor Perer and we decided that given time constraints I should focus on the most achievable visualization. To me, that meant creating a series of tabs that each contained one single view of the data. I tried coding one very simple representation of “words of interest” from 1999 to 2006.



Tab One

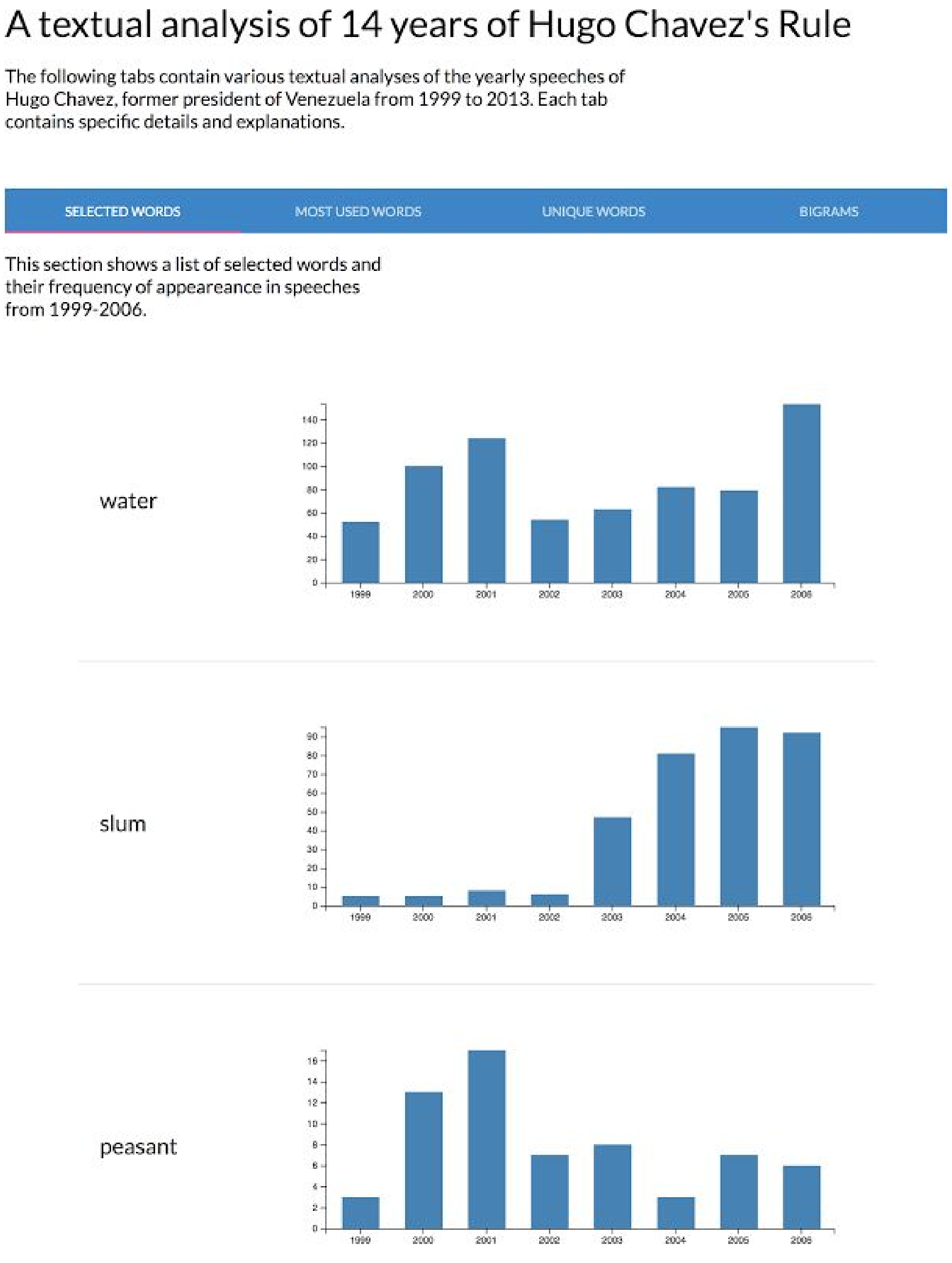
This is an example tab.

You can put any sort of HTML or react component in here. It even keeps the component state!

Graphs of the frequency of appearance of six different select words from 1999 to 2006

**Final Implementation**

After more iteration and some work to fix the formatting of the graphs, I ended up with the current version. The whole page's use is not too complex, it requires users to click through different tabs to access four visualizations: a list of selected words and their use over 8 years, most commonly used words per year, unique words per year, and a list of likeliest bigrams per year.

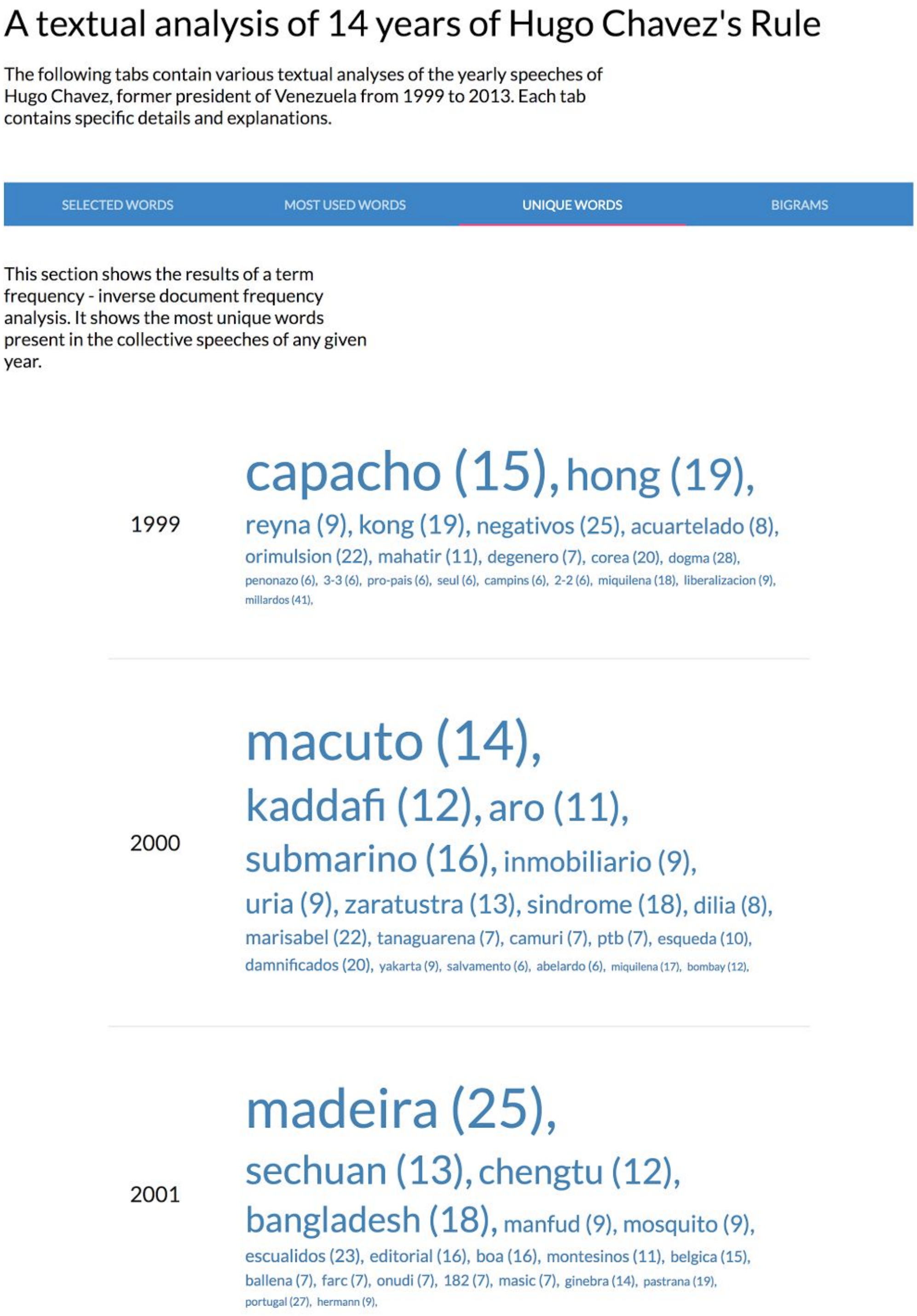


A textual analysis of 14 years of Hugo Chavez's Rule

The following tabs contain various textual analyses of the yearly speeches of Hugo Chavez, former president of Venezuela from 1999 to 2013. Each tab contains specific details and explanations.

This section shows a list of selected words and their frequency of appeareance in speeches from 1999-2006.

A first tab showing the frequency of selected words



## A textual analysis of 14 years of Hugo Chavez's Rule

The following tabs contain various textual analyses of the yearly speeches of Hugo Chavez, former president of Venezuela from 1999 to 2013. Each tab contains specific details and explanations.

This section shows the results of a term frequency - inverse document frequency analysis. It shows the most unique words present in the collective speeches of any given year.

1999

capacho (15), hong (19),
reyna (9), kong (19), negativos (25), acuartelado (8),
orimulsion (22), mahatir (11), degenero (7), corea (20), dogma (28),
penonazo (6), 3-3 (6), pro-pais (6), seul (6), campins (6), 2-2 (6), miquilena (18), liberalizacion (9),
millardos (41),

2000

macuto (14),
kaddafi (12), aro (11),
submarino (16), inmobiliario (9),
uria (9), zaratustra (13), sindrome (18), dilia (8),
marisabel (22), tanaguarena (7), camuri (7), ptb (7), esqueda (10),
damnificados (20), yakarta (9), salvamento (6), abelardo (6), miquilena (17), bombay (12),

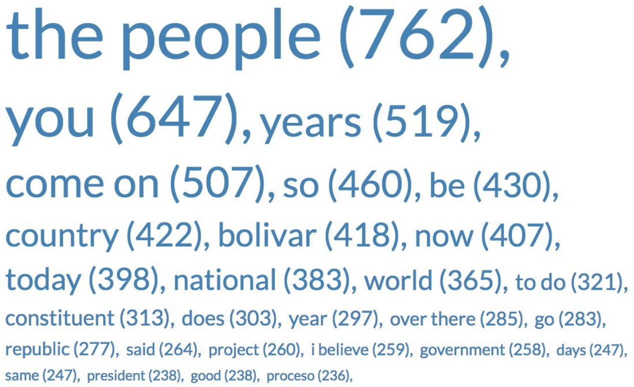
2001

madeira (25),
sechuan (13), chengtu (12),
bangladesh (18), manfud (9), mosquito (9),
escualidos (23), editorial (16), boa (16), montesinos (11), belgica (15),
ballena (7), farc (7), onudi (7), 182 (7), masic (7), ginebra (14), pastrana (19),
portugal (27), hermann (9),

Another tab showing the results of the tf-idf analysis

A few design decisions I made guided by what we learned in class:

- The choice to aggregate speech data in bins that represented years was based on the idea of simplification. Some kind of bar chart based on individual speeches or even a monthly view into the words appearing in speeches would have been too noisy and large.
- I relied on the idea that small multiples can help communicate complex data. Instead of somehow combining all the data I had per year into one single chart, I decided to separate them into different visualizations.
- Instead of using a traditional word cloud, which can make it hard to distinguish smaller differences between words, I organized words in order of frequency and also added a frequency count on the side:



the people (762),
you (647), years (519),
come on (507), so (460), be (430),
country (422), bolivar (418), now (407),
today (398), national (383), world (365), to do (321),
constituent (313), does (303), year (297), over there (285), go (283),
republic (277), said (264), project (260), i believe (259), government (258), days (247),
same (247), president (238), good (238), proceso (236),

I certainly deviated a bit because of time constraints. For example, I didn't get a chance to work on as much interaction as I would have wanted, but I was hoping to allow for users to hover over certain words in the word clouds or bars in the bar charts to gain more information about a given data point.

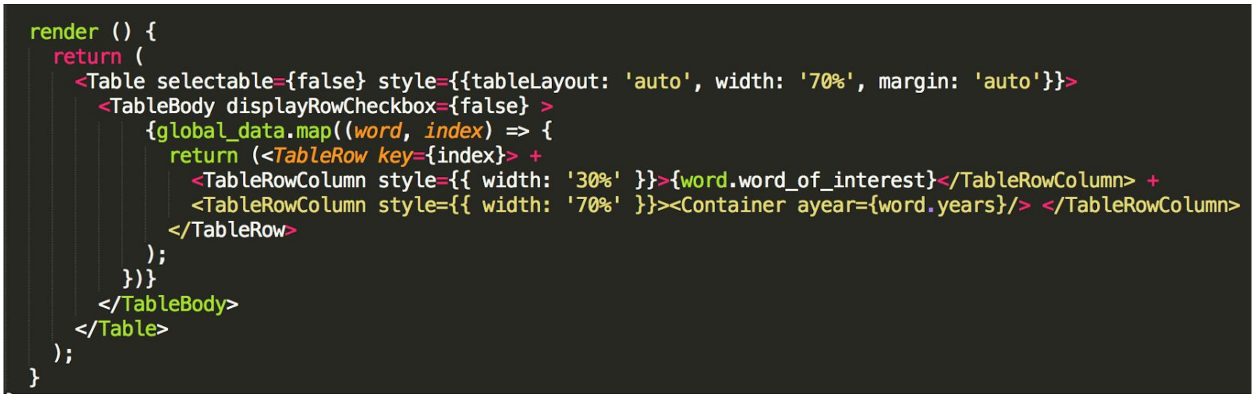
### Coding Details

Given how many repetitive elements I had on each tab of the visualization, I was curious to see if I could combine React and D3 to make creating components more reusable and maintainable. Having used React in the recent past I was also simply curious to push my web skills and see what the state of the art was in terms of combining these two.

It *is* a little tricky, given that both D3 and React control the DOM. After reading through several blogs and testing out a few different libraries, I discovered a third party library (react-faux-dom) that allows you to parse D3 code through

React while still relying on D3 for setting up new elements in the DOM and performing math.

In the end, I was able to write code like so to create, for example, the first tab with Words of Interest. I can essentially use an array to iterate through the data ("global_data") and and create a table row ("TableRow" component) for each one.

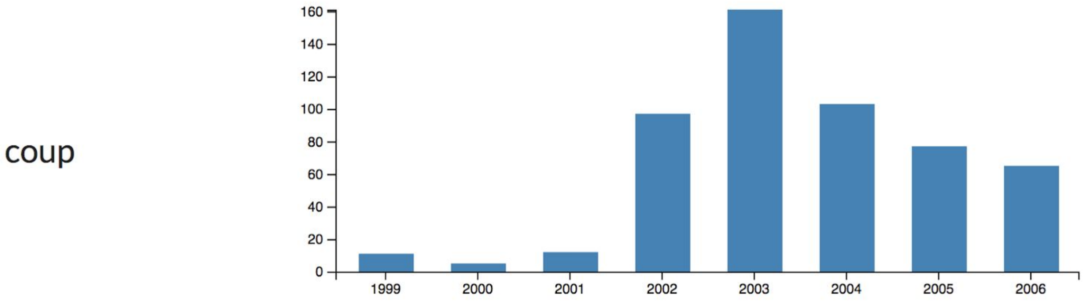


```
render () {
  return (
    <Table selectable={false} style={{tableLayout: 'auto', width: '70%', margin: 'auto'}}>
      <TableBody displayRowCheckbox={false} >
          {global_data.map((word, index) => {
            return (<TableRow key={index}> +
              <TableRowColumn style={{ width: '30%' }}>{word.word_of_interest}</TableRowColumn> +
              <TableRowColumn style={{ width: '70%' }}><Container ayear={word.years}/> </TableRowColumn>
            </TableRow>
          );
        })}
      </TableBody>
    </Table>
  );
}
```

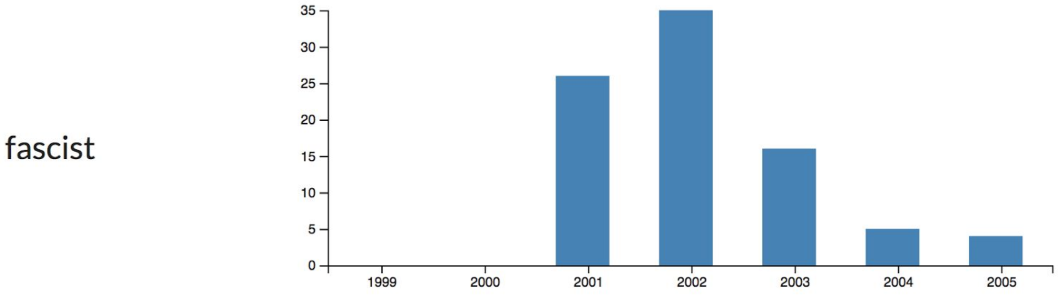
# Evaluation

A few insights I gained from looking at the data:

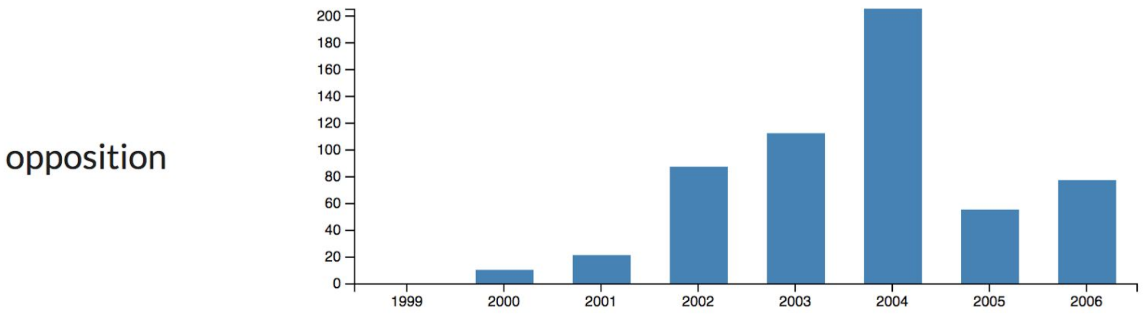
*1. On a basic level, I wanted to see if Chavez's language clearly reflected important political events. One would imagine that they would, but it's reassuring to see it in the data. Looking at how often "coup" appears it becomes clear he is referring to an attempt to unseat him from power in 2002.*



The opposition in Venezuela attempted a coup in 2002. It failed and heralded a new period of radicalization for Chavez.

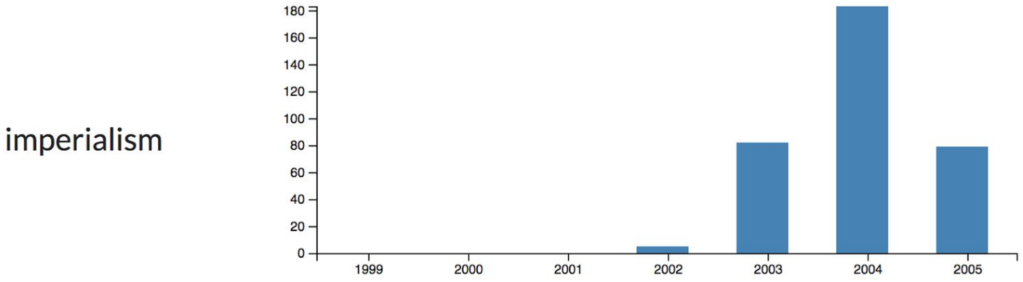


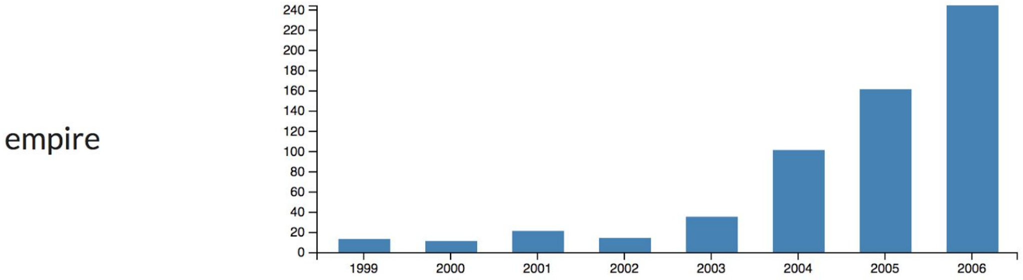
Looking at the word "fascist" in this chart, I'm curious to dig deeper into how Chavez is using it. Is he referring to the opposition and the events that led to the coup? As a researcher, this would be a starting point for further investigation.



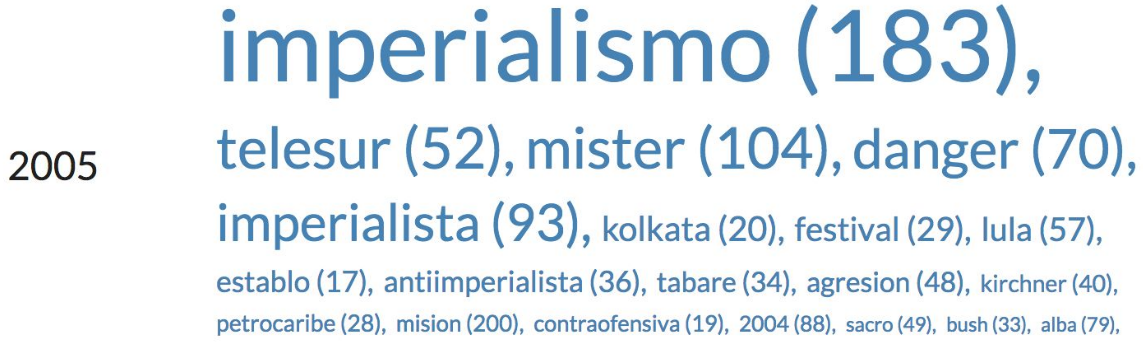
Why is there a dip in the word opposition in 2005? Another starting point for investigation.

*2. Chavez's starts to reference the U.S. a lot more starting in 2003, reaching a peak in 2005. This might make sense given he became more radicalized after the 2002 coup and started blaming the U.S. for more of Venezuela's problems.*





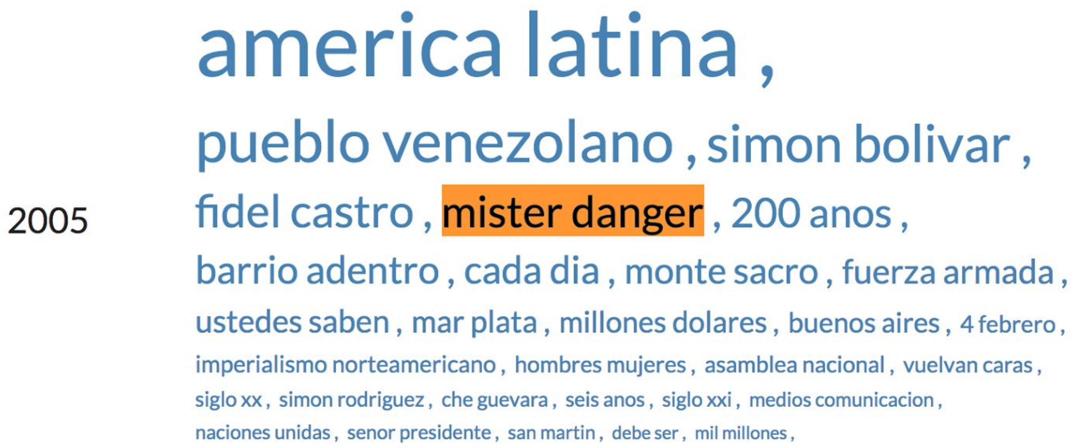
Notice that “imperialism” and “empire” (which is actually referring to the U.S.) are virtually not mentioned before 2002.



2005

imperialismo (183), telesur (52), mister (104), danger (70), imperialista (93), kolkata (20), festival (29), lula (57), establo (17), antiimperialista (36), tabare (34), agresion (48), kirchner (40), petrocaribe (28), mision (200), contraofensiva (19), 2004 (88), sacro (49), bush (33), alba (79),

For example: Words like “imperialism” and “mister danger” (a reference to Bush) appear in the tf-idf analysis in 2005.

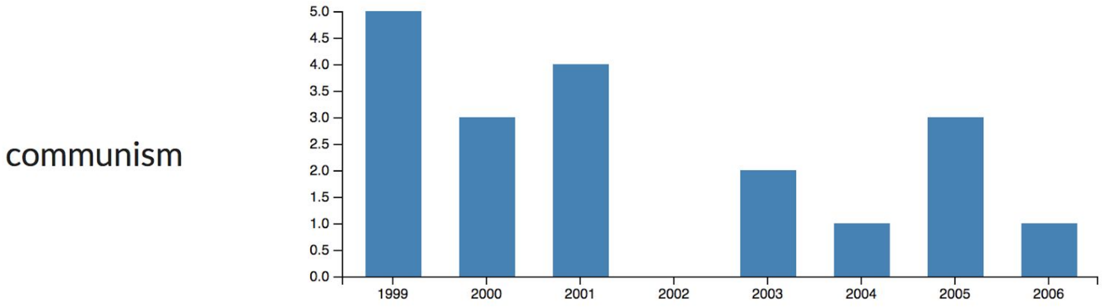


2005

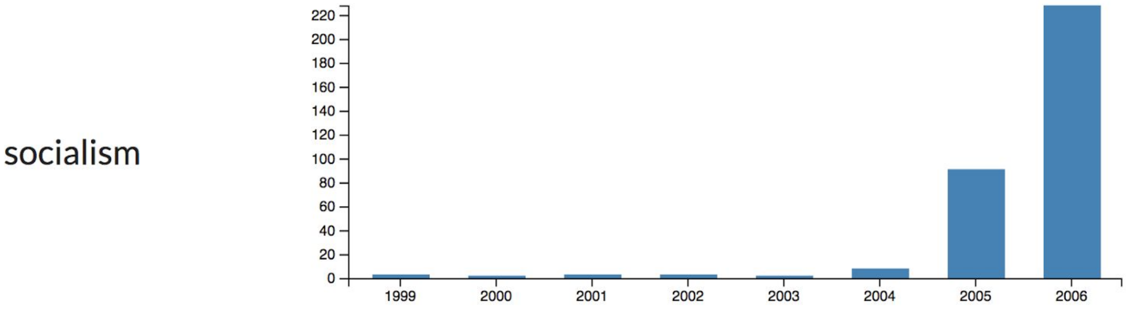
america latina , pueblo venezolano , simon bolivar , fidel castro , mister danger , 200 anos , barrio adentro , cada dia , monte sacro , fuerza armada , ustedes saben , mar plata , millones dolares , buenos aires , 4 febrero , imperialismo norteamericano , hombres mujeres , asamblea nacional , vuelvan caras , siglo xx , simon rodriguez , che guevara , seis anos , siglo xxi , medios comunicacion , naciones unidas , senor presidente , san martin , debe ser , mil millones ,

Bigrams like “mister danger” appear in 2005.

*3. Chavez was careful with his use of “communism” and “socialism”. Furthermore, references to socialism really begin in 2005. This was surprising to me, but in retrospect makes sense: Chavez radicalized slowly - it took him years to build up to his idea of a socialist bolivarian revolution.*

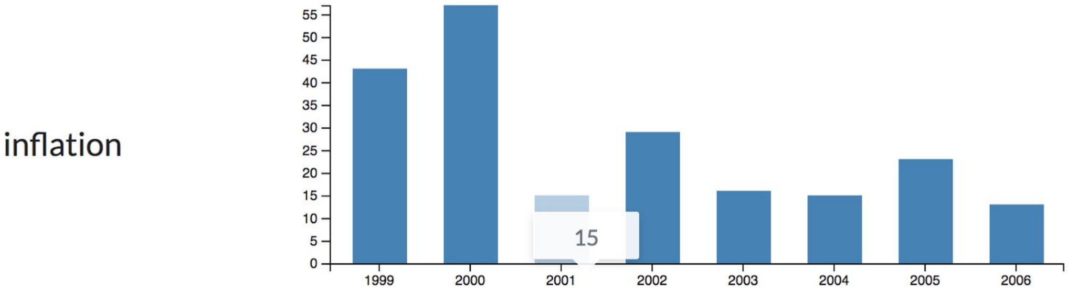


Notice that communism appears less than 5 times per year.

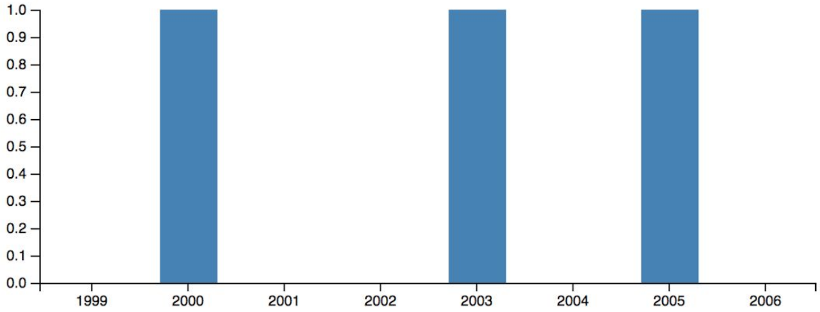


Socialism starts appearing in his speeches in 2005.

*4. Chavez avoids talking about all the ills that start affecting the country during his rule and have hit a breaking point today. Venezuela has the highest inflation in the world today and it's capital Caracas is the world's most dangerous.*

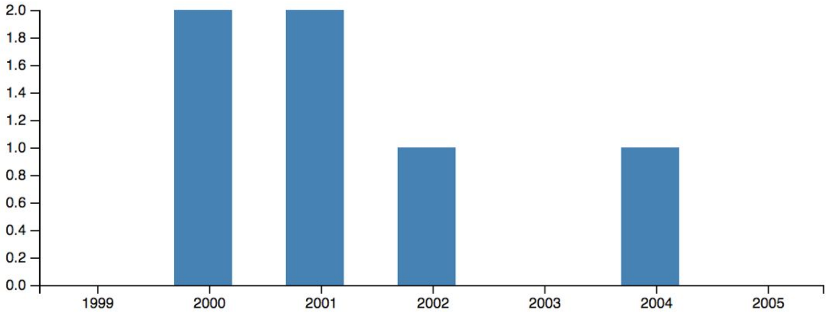


Talk of inflation is limited starting in 2001 scarcity



Scarcity is mentioned 1 single time in 2001, 2003, and 2005.

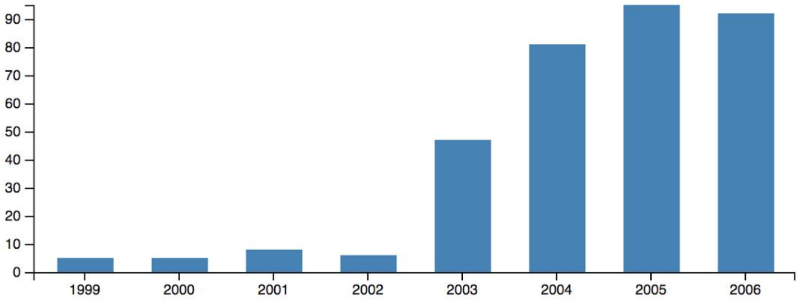
crime



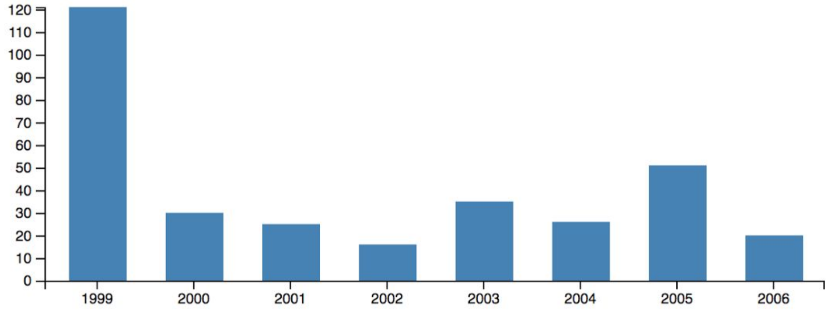
Crime mentioned between zero and two times in different years.

*5. Two open questions I'd investigate further by digging into individual speeches: a) why are slums only mentioned starting in 2003? What happened? and b) did Chavez use the word "crisis" so much in 1999 to describe the nation he was inheriting? Or for some other reason?*

slum



crisis



As a first version, I'm satisfied with the simplicity and use of my visualization. There were some insights I was satisfied to see reflected in the data (i.e. knowledge about Venezuela I already knew) and other charts that were more curious and led me to wonder what might be going on. As a researcher, these are precisely the kinds of insights that would allow me to dig deeper into such a large dataset.

## Improvements and Next Steps

I have many ideas for how to improve the project and add functionality. One obvious first step is to parse the rest of the data from 2007-2013. I simply didn't have enough time to do so the first time around.

Next steps around text analysis:

- Perform a sentiment analysis
- Add a few more stopwords to clean up the data: I believe the spanish stop words list used by NLTK is pretty limited. It's likely not as good as the english one. I can certainly improve some of my results by further adding common but not very meaningful words to my list.
- Verb visualization: as mentioned above, I coded a parts of speech parser to extract only verbs from different speeches. I'm not sure if this data is actually useful, but it would be interesting to explore

Next steps around visualization:

- Test more ways to bring the data together. I'm currently using tabs to separate the different visualizations, but maybe there's a different way
- Tell a story with the data
- Add a timeline view