Tone of Voice Predicts Political Attitudes

Evidence from U.S. Supreme Court Oral Arguments

Yassine Kadiri
New York University
Center for Data Science
yk1859@nyu.edu

Zsolt Pajor-Gyulai
New York University
Courant Institute of
Mathematical Sciences
zsolt@cims.nyu.edu

Thomas Leble
New York University
Courant Institute of
Mathematical Sciences
tl78@nyu.edu

Elliott Ash
ETH Zurich
Center for Law & Economics
ashe@ethz.ch

Daniel L. Chen
Toulouse School of Economics
daniel.chen@iast.fr

ABSTRACT

Can we guess the political ideology of people based on the way they speak? We investigate this question in the context of the hearings of the Supreme Court. The speakers are judges and attorneys, arguing over different cases. We measure speaker ideology using political donations data. We find that basic phonetics data together with linear learning models yield predictions with an accuracy of approximately 73%, beating random guessing. Tone of voice is predictive of political attitudes.

ACM Reference Format:

Yassine Kadiri, Zsolt Pajor-Gyulai, Thomas Leble, Elliott Ash, and Daniel L. Chen. 2018. Tone of Voice Predicts Political Attitudes: Evidence from U.S. Supreme Court Oral Arguments. In *Proceedings of (CELS).* ACM, New York, NY, USA, Article 4, 8 pages. https://doi.org/10.475/123_4

ACM ISBN 123-4567-24-567/08/06.
https://doi.org/10.475/123_4

1 INTRODUCTION

Recent work in social sciences has documented that written and spoken text is predictive of political attitudes (e.g. Gentzkow et al, 2017). In the case of the Supreme Court, Sen et al. (2018) show that emotional arousal is predictive of voting.

Estimating a judge's ideology or that of a legal representative's arguing in front of the court is not an easy task as the political chess is such that a single person will be considered conservative on some subjects and liberal on others. Such misinterpretation of a candidates ideology happened to previous presidents who nominated justices that ended up voting against them often causing significant political damage.

Therefore, having a more precise appreciation of a judge's or a lawyer's ideology can be critical when the time comes to for a president to appoint a new judge. Intuitively, given the fact that justices are nominated after a long history of practice, this means we technically have enough data to assess their ideology upon nomination. At the same time, predicting speakers' ideology based on their speech turns out to be not such an easy task.

In this article, we take a new approach by using TextGrid data. This data type which links speech and pronunciation, providing phonetics data for each word. Based on this phonetics data we try to clasify one's ideology as conservative or liberal

The results demonstrate that the way speakers pronounce certain words does indeed yield some predictive power which in itself is suprising as we would expect speakers to have a more neutral intonation in order not to reveal their ideology. Also, some words that we will qualify as ideologically charged tend to provide more predictive power with less data showing that not only the pronunciation and energy put in a word is important but so is the word itself.

2 RELATED WORK AND HYPOTHESIS

We have seen that estimating the ideology of potential justices is a critical issue for the President's party when the time comes to appoint new justices. The literature already deals with estimating the ideology based on text data and on non verbal expressions, even though the measurment and the very definition of "ideology" are subject to debate.

Previous work has shown that political judgement, i.e. the assessment of one's political affiliation by voters, was influenced by how feminine or masculine politicians were perceived [1] and more surprisingly also based on their faces [2] depending on how powerful or warm they were perceived. Those two articles confirm an intuition which is that our non-verbal expressions betray to some extent the political views we might have.

So the form is relevant to assess one's political ideology but so is the substance. Previous work has been done on social media in order to assess users' political orientation based on their tweets [4] but the performance was higher when training a Support Vector Machine on the hashtags rather than on the full tweets' text. Moreover, estimating the users' ideology was easier when using network properties and studying the spread of information in the users' networks. The article exploits text but also network information that users create when they tweet in order to classify groups of users into a political orientation. However, classifying a single person's ideology based simply on text is trickier than expected [3] as a tricky part is to capture the features which allow to classify users. On Twitter, those features change depending on the type of user we're considering, a consequence of this being that a classifier trained on either type of users defined in [3] will performed poorly on the other type of user, performing even worse than random guessing.

Therefore, simple methods prove to be not predictive of ideology when based on text. However, [5] was able using a Recursive Neural Network at the sentence level to estimate the ideology of a speaker based on the words he uses. This article even shows that this method outperforms the more classical "bag of words" method. To some extent, the work done on sentiment analysis [6][7] allowed to see that semantic structures we use can be understood better and we can detect underlying bias, if there is, and this is partly what is behind the intuition in [5].

Given all of this, our hypothesis is that analyzing audio or phonetics data might yield information as the way public speakers pronounce words might be influenced by their ideology but also by their non-verbal attitude at the moment they speak their words. The combination of those two factors which have proven to convey speakers' ideology could therefore lead to audio or phonetics data yielding indeed some predictive power, and this is what we will prove.

3 DATA AND METHODS

3.1 Datasets

We used two datasets in our work: (1) the Oyez dataset [1] which contains audio recordings of Supreme Court hearings between 1998 and 2013, together with a textual transcription of the arguments and (2) Stanford's DIME dataset[2]. More details about both dataset and their format can be found in the appendix.

3.2 Data Preparation

3.2.1 *Phonetics data.* .

Gather speakers data

In the Oyez dataset, we look for the phonetics data for speakers at the Supreme Court of The United States, process the data to store it in a convenient format. We also store a list of all words which appear in our data and how much they appear.Âă

Getting one's ideology

To do so, we use the DIME dataset developed by Adam Bonica[9]. We preprocess this list in order to get for each speaker his ideology. The details of how we did this can be found in appendix. We run our method on two datasets: the 2008 political campaign (including State and Local elections) and the combined Presidential campaigns (all Presidential elections since the mid-70's)

[1] https://www.oyez.org/about

[2] https://data.stanford.edu/dime

and we take the average (which is admittedly not a proper barycenter). We then get an ideology with values between 0 (“Pure" Democrat) and 1 (“Pure" Republican). This way, we obtain an ideology label for around 500 speakers over approximately 900.

Creating training data

With the TextGrid files and the DIME dataset, we extract for each word and vowel in that word the phonetics data for each speaker along with the ideology for each speaker. In order to avoid overfitting to the most frequent speakers, we average all phonetics data for each speaker in order to have only distinct speakers and how their average phonetics data for that vowel in that word. We however remove speakers for which the ideology is undefined. The resulting files are distinct files for each word and vowel in that word. This gives files of the form `Word_Vowel_Position.txt` where position is the position of that vowel in that word (for example `Stop_OH_1.txt`). This position argument was introduced for reasons detailed in the appendix. The main idea is that for a same word, depending on the speaker, one same vowel can be pronounced differently. In each of those files, the entries are lists containing the ideology, how many times that speaker has pronounced that word and vowel and the phonetics data for that vowel in that word. Details of this method and its results can be found in appendix.

4 MODELING

For the phonetics data, we worked with linear models. The main models we used are Lasso and Ridge Regression, along with Logistic Regression. This choice comes from the size of our datasets. Given how we built the files for each word, we find ourselves with files with at most 450 instances. Therefore, we’re dealing with small data and we need to be careful with overfitting so this is why we chose regularized linear models. In order to choose the regularization strength, the idea is to test several values for λ for each regressor and choose the value for which the error is minimized or equivalently the value for which the score is maximized. Then, we try to predict the ideology of speakers. We first took this problem as a regression task but shifted to a classification task given the size of the dataset which wouldn’t allow a precise enough prediction. We set the threshold to 0.5 i.e. if the regressor predicts an ideology greater than 0.5 we set the prediction to 1 and else to 0. The validation metric we then chose was the accuracy of our prediction along with the F1-score in order to have a decent appreciation of our performance.

The idea here is to have a regressor ready for each word and each vowel for that word so that when given a new speaker, we can predict his ideology based on how he pronounces the words we selected. Given how we built our dataset, we can not overfit to a single speaker as each dataset contains only distinct speakers. Also, in order to avoid overfitting and make the most of the small datasets we have, we decided to make a cross-validation for each of the regressors.

For now, we have run our algorithm on all words who occur with more than 4 distinct speakers. This choice of values can seem arbitrary but in our preliminary work, we were able to see that past 20 distinct speakers, the accuracy obtained is close enough to its value with bigger sets. Therefore, the main hyperparameter here is the number of distinct speaker. However we choose to go for a smaller number in order to witness how fast the convergence of our metrics is and most of all to which value it converges to.

5 ANALYSIS

For the phonetics data, the scores were really poor and we were able to increase them by removing some features (such as ’Count’ which the number of times a speaker had spoken and that we thought would be predictive). Once this was done, we got better scores, around 0.6 and could adequately choose our regularization strength for each regressor and for each file. For each file (`word_vowel.txt`) we therefore choose the appropriate λ for each regressor and for each of the 5 folds of our cross-validation, predict the ideology of the 20% of speakers from that fold. Then we compute the accuracy and f1-score for each of the 5 predictions and average it, for each regressor. We decided to plot the results as a function of the size of the file for which we made the predictions. The idea here is, given the small data we have, to be able to estimate how our predictions perform ’asymptotically’ when the data becomes bigger.

The results are given in figures 1 and 2

It seems that the Lasso, Ridge and Logit ’classifiers’ have similar results and asymptotically reach an accuracy of 73%. We notice that for small datasets, the results have really high variance, as expected, but as

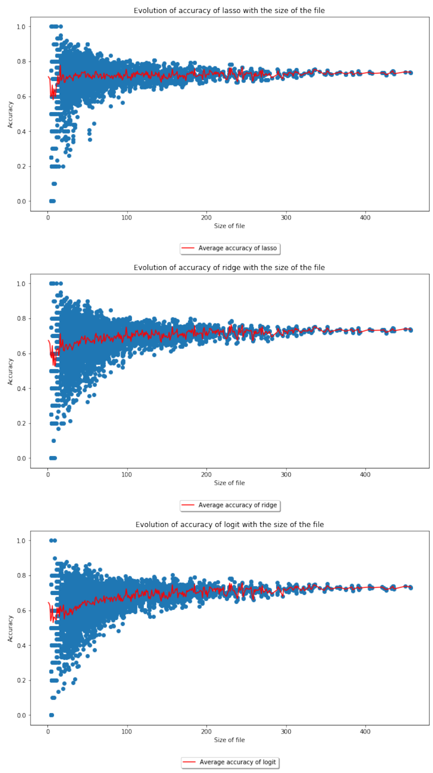

Figure 1: Results for accuracy with lasso, logit and ridge classifiers

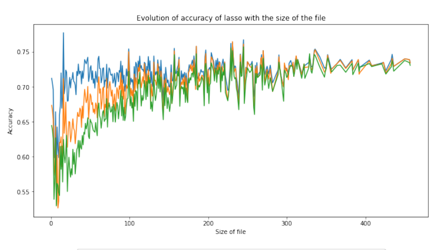

Figure 2: Comparing average accuracy on all words for the three classifiers

the size grows, the accuracy increases and converges to a final value. The figures show that this convergence is quite fast as for 200 words we almost reach the final value and the variance is low. However, we see that lasso's variance decreases faster than other classifiers. On average, all classifiers perform above 55% accuracy as figure 2 shows. In figure 3, we see that indeed the

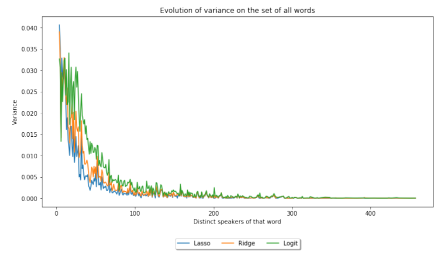

Figure 3: Comparing the variance for all classifiers

variance decreases faster for lasso and that on overall, all classifiers converge fast towards zero variance.

The aim now is to see which words are the most predictive as we might find some words that aren't really neutral as in some situations there are no neutral alternatives to a word as for example what liberals call "estate tax" is called "death tax" by conservatives [5][10]. So what we did is take the best performers for each classifier and tried to compare how they perform with other classifiers. The plots in figure 4 show that the vast majority of words perform above 60% accuracy for each classifier however, one good performer for lasso isn't necessarily a good performer for ridge or logit and it's the same the other way around.

Also, another aim of our study was to see exactly which were the best words to predict one's ideology and we see from the plots in figure 4 that there might not exist any "universal" best performers. However, we can see which are the best performers for each classifier. Taking words that appear a lot won't help as those are generally too common to convey any kind of ideology. However, one could argue that if those frequent words perform well, they should be used to predict ideology and that taking words that don't appear too often would bring us to high variance and therefore no generalization would be really possible.

Yet, we must keep in mind that we don't actually deal with words but with tuples of the form (Word,vowel,position of the vowel) meaning that low number occurences doesn't mean low number occurences of the word itself but for the vowel. This will become more clear when displaying the best performers but the idea is that a tuple (word,vowel,position of the vowel) which doesn't appear too often is more likely to betray an accent, a

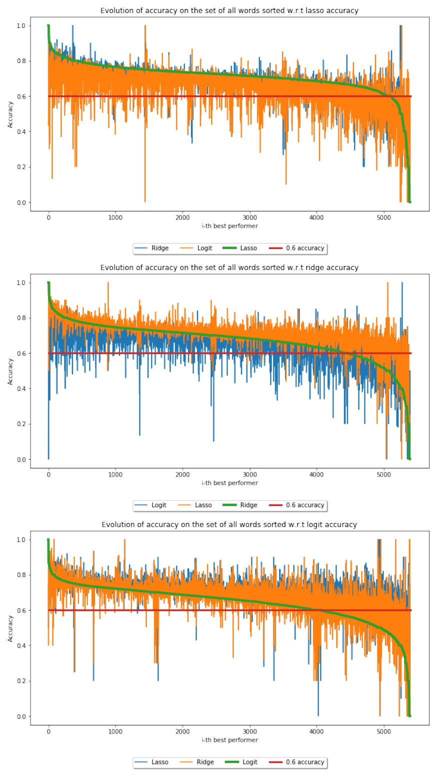

Figure 4: Comparing the accuracy of the best performers between classifiers

Table 1: Words-vowels that perform best with lasso classification, for each word we display the number of distinct speakers, the average ideology of those speakers. First we have the best performers, and then we have the words appearing more than 110 times

word	Acc	Speakers	Ideology
ENORMOUS_IY_1.0	1.0	11	0.27
COMMISSION_AH_2.0	1.0	4	0.5
MITIGATING_IH_3.0	1.0	6	0.33
ADEQUATE_EY_3.0	1.0	16	0.25
OFFICIALS_AH_2.0	1.0	5	0.4
PARTICULARLY_AA_1.0	1.0	9	0.44
SOUTER_AW_1.0	1.0	6	0.33
ANYTHING_IH_2.0	0.95	16	0.13
ACKNOWLEDGE_AE_1.0	0.93	12	0.25
RECONSIDERATION_EY_5.0	0.93	16	0.19
WERE_IH_2.0	0.93	13	0.15
CREATED_IY_1.0	0.81	135	0.19
CREATED_EY_2.0	0.81	134	0.19
CREATED_AH_3.0	0.81	134	0.19
SUGGEST_EH_2.0	0.79	185	0.21
SUGGEST_AH_1.0	0.79	185	0.21
PRESENTED_AH_3.0	0.79	180	0.21
PRESENTED_EH_2.0	0.79	180	0.21
WHAT_AH_2.0	0.79	124	0.22
SIGNIFICANT_IH_2.0	0.79	158	0.21
SIGNIFICANT_IH_1.0	0.79	158	0.21
BEFORE_IY_1.0	0.79	147	0.21

specificity that betrays one's origins or level of education or many other variables which themselves condition one's political affiliation. An interesting problem would be to see which variables pronunciation can betray and how those variables are themselves predictive of one's ideology.

This is why we didn't set any threshold regarding the number of distinct speakers speaking a (Word, vowel) and just decided to look to the best performers right away and also see which ideology is more discriminated by our algorithms. We then also displayed the most frequent words and their performance. Turns out, as table 1, 2 and 3 show, that our algorithms are more likely to perform better with words pronounced by justices/lawyers that are democrats. We display for each classifier the 10 most predictive words without threshold and the 10 most predictive words that appear more than 110 times (arbitrary threshold)

We therefore see that even though the 'limit' is around 73% for our 'classifiers', the best performers reach around 78-80% accuracy while some even reach 100%. This mark of 100% might sound too good to be true and obviously in certain cases it is, however, in some situations, it might just mean we spot an accent or a speech pattern strongly correlated with an ideology, republican or democrat. Also, we can see that our best performers are associated with speakers that are, on average democrats. As it is a binary classification task, it also means that our algorithms discriminate well against republicans. The confusion matrix for our best model (Lasso Classification) proves this statement, however it shows that our model is more prompt to declare a speaker as Democrat. This comes from the fact that most of the speakers are closer to being democrats than republicans. Nevertheless, the good performance on the words used for the confusion matrix simply shows that

Table 2: Words-vowels that perform best with ridge classification

Word	Acc	Speakers	Ideology
BECAUSE_AO_4.0	1.0	4	0.5
SEVEN_EH_3.0	1.0	7	0.29
ENORMOUS_IY_1.0	1.0	11	0.27
BELIEVE_AH_3.0	1.0	4	0.5
IMPORTANT_AH_2.0	1.0	4	0.5
ADEQUATE_EY_3.0	1.0	16	0.25
OFFICIALS_AH_2.0	1.0	5	0.4
SOUTER_AW_1.0	1.0	6	0.33
ANYTHING_IH_2.0	0.95	16	0.13
ACKNOWLEDGE_AE_1.0	0.93	12	0.25
RECONSIDERATION_EY_5.0	0.93	16	0.19
PRESENTED_AH_3.0	0.8	180	0.21
CREATED_EY_2.0	0.80	134	0.19
CREATED_AH_3.0	0.80	134	0.19
SIGNIFICANT_AH_4.0	0.80	157	0.21
SUGGEST_EH_2.0	0.79	185	0.21
SUGGEST_AH_1.0	0.79	185	0.21
COUNSEL_AW_1.0	0.79	160	0.22
CREATED_IY_1.0	0.79	135	0.19
WHAT_AH_2.0	0.79	124	0.22
SIGNIFICANT_IH_2.0	0.79	158	0.21
EXCUSE_IH_1.0	0.79	127	0.21

Table 3: Words-vowels that perform best with logistic classification

word	Acc	# Speakers	Ideology
IMPORTANT_AH_2.0	1.0	4	0.5
INDEPENDENT_AH_1.0	1.0	9	0.44
PARTICULARLY_AA_1.0	0.9	9	0.44
ARTICULATE_EY_4.0	0.88	24	0.17
VIOLENCE_AY_1.0	0.87	22	0.14
VIRTUALLY_IY_4.0	0.87	17	0.18
A_AH_4.0	0.87	12	0.21
MATERIALITY_IY_3.0	0.87	17	0.24
APPELLATE_EY_3.0	0.87	14	0.14
WITH_AH_1.0	0.87	15	0.27
EXAMINATION_IH_1.0	0.86	52	0.15
SIGNIFICANT_AH_4.0	0.80	157	0.21
CREATED_EY_2.0	0.79	134	0.19
SUGGEST_EH_2.0	0.79	185	0.21
SUGGEST_AH_1.0	0.79	185	0.21
CREATED_IY_1.0	0.79	135	0.19
BEFORE_IY_1.0	0.78	147	0.21
UNDERSTAND_AH_1.0	0.78	231	0.22
CREATED_AH_3.0	0.78	134	0.19
SIGNIFICANT_IH_3.0	0.78	157	0.21
PRESENTED_AH_3.0	0.78	180	0.21
UNDERSTAND_AE_3.0	0.78	229	0.22

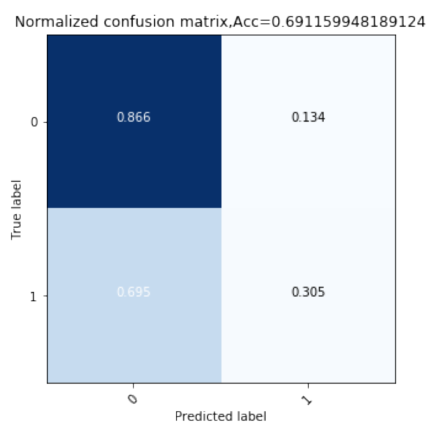

Figure 5: Confusion matrix for Lasso for our top 362 performers

most of those words are strongly correlated with being democrat. Finally, we also see that some words occur as top performers with all of our classifiers.

6 CONCLUSION

We found that simple learning models worked fairly well given the small size of our final data sets and the errors that may have been introduced along the way (e.g. the possibility that the ideology of a speaker had been computed by taking into account the donations of an homonymous attorney, the fact that we did not take gender or region of origin into account), showing some predictive power. It "detects" *ideology* among speakers, performing very well with words associated with a majority of democrat speakers. A perspective for future work would be to see to which extent we detect ideology and if, in some cases we aren't actually spotting more basic characteristics (e.g. masculinity, low tone of voice, geographic accent) that would be correlated with ideology.

APPENDIX

A DATASETS

A.1 The Oyez Dataset

This dataset contains audio recordings of Supreme Court hearings between 1998 and 2013, together with a textual transcription of the arguments. The audio data has been parsed by a phonetics software (Praat) to extract the most relevant phonetics data. The final result consists of Audio files containing the hearings and TextGrid files containing the breakdown of the hearing into speakers, sentences, words, syllables, and for each syllable the relevant phonetics data (called the **formants**) as to how this vowel was pronounced.

To get a better sense of how this dataset works, we'll consider an example. For a given case we have an associated date/identification number. For example, let's consider case `2006_04_1350`. For this case, we have an associated text-grid file `2006_04_1350.TextGrid`. This file contains the breakdown by sentences/paragraphs. It contains a number N of Âń items Âż which is indicated at the beginning of the file as the size parameter (here, $N = 11$). Each item corresponds to one person speaking (here, items will range from `01` to `11`). For any given hearing, there are $4M$ files, where M is less than or equal to N - the number of items (or size) as above. The factor 4 comes from the fact that we have, for each speaker, 4 corresponding files. For `2006_04_1350` for example, $N = 11$ but $M = 4$. Let's say we want to study speaker K, then the relevant files for us are those named `2006_04_1350_sK_norm.txt` where K is the index of the speaker. They contain the breakdown of speaker K's speech by word/vowels, together with the relevant phonetics information for each vowel (the duration, the two first formants, and more formant data).

A.2 The DIME dataset

The Database on Ideology, Money in Politics, and Elections dataset [9] collects "over 130 million political contributions made by individuals and organizations to local, state, and federal elections spanning a period from 1979 to 2014." We extracted an abridged version of this dataset, containing only the donors who have declared to be working in the judicial world. Then, we used the normalized Martin-Quinn[8] score [3] as an indicator of ideology for the justices (judges) at the Supreme Court. This score allowed us to label a big part of our data as based on their donations, this score allows to retrieve an estimate of justices and lawyers' ideology at the SCOTUS.

[3]http://mqscores.lsa.umich.edu/measures.php

B DATA METHODS

B.1 Gathering a list of speakers and words spoken

This is done with the TextGrid files described before and which breakdown into two files. From the first file (the .TextGrid), we find the name of speaker K, and we then gather the phonetics data associated to this speaker by going through the corresponding `_norm.txt` file. In an exploratory phase, we figure out the words most pronounced by our speakers, as well as a list of all the speakers that appear. The number of speakers is of order 1000.

B.2 Getting speakers' ideology

The DIME dataset lists the financial contributions to political campaigns in the USA. Some basic difficulties appear however. The database is very large, with several dozens of millions of entries, and a naive thorough search takes too much time if it has to be done for around 1000 speakers. Also, some common names may appear several times which brings another problem: how to identify a speaker in the donation list? Finally, we also encounter another less obvious issue as it is fairly common that a given person will give money to both parties.

In response to those issues, we choose to go through the list of donation exactly once, and only keep the entry if the "job/activity" contains a word relevant for us: lawyer, justice, attorney, justice, law. This effectively reduces the size of the list by a factor 10 to 100. Then for each speaker, we go through the reduced list and try to match the first/last name with the first/last name entry of the donation. Here we make the following bet: there is only one person with a given first and last name who happens to work as a lawyer. A more careful procedure would involve checking the middle name (not always present in both tables) or the addresses and listing the possible collisions.

If a speaker is not present, his/her ideology is set to "Undefined". Otherwise, we take the average amount

$$\textbf{Ideology}(\text{speaker}) = \frac{\text{Donation for Republican candidates}}{\text{All donations of the given speaker}}$$

Hence an ideology of 0 corresponds to a pure Democrat and ideology of 1 corresponds to pure Republican.

B.3 Creating the training data

We use the TextGrid files to extract, for each speaker, a list of words they pronounced along with the formants' data for each word. The resulting file is a dataframe whose entries are of the type

word,speaker,list of syllables,formants data for each syllable

We then use the DIME dataset to find the ideology for each speaker. We also gather the judges "ideology" from their Martin-Quinn score.

Then, we reformat the data produced by our data gathering scripts. We form a dictionary whose keys are tuples (speaker, word, vowel, position_in_the_word[4]) and values are the phonetics data, namely a list

$$[\text{count}, f_1, f_2, \text{duration}, \text{formant1}, \text{formant2}]$$

where both f_1 and f_2 are the two main frequencies recorded when that word was spoken and formant1 and formant2 are themselves lists containing 3 frequencies each. The script also populates a list of all the speakers for whom we have phonetics data. This will help us compare and see if for those speakers we have an ideology data.

We then average the formants' data over all occurrences in the dictionary. (This is why we introduced count in our dictionary). The aim here is to avoid beforehand the situation in which the models would overfit to one single speaker. As some speakers speak way more than others, this would skew our models towards some speakers if those happen to pronounce a word more than others. Also, this will help create datasets with distinct speakers which will provide independent training-testing sets when we'll split. Finally, we get the ideology for the speaker and matching both dictionaries. We remove entries for which we don't have ideologies in order to avoid missing values.

This way for each entry

$$(\text{word, vowel, position\_in\_the\_word})$$

we produce a file containing phonetics data for each distinct speaker who pronounced that word and vowel, namely lists

$$[\text{ideology}, \text{count}, f_1, f_2, \text{duration}, \text{formant1}, \text{formant 2}]$$

The names don't appear anymore as we mostly are interested in the ideology behind a name.

[4]This last key position_in_the_word was introduced as for some words, speakers pronounce them differently which ends up in the formants data attributing differents "heard" vowels for the same vowel, for example the O in STOP can be heard as an AH or an OH depending on the speaker

REFERENCES

[1] Carpinella, C. M., Johnson, K. L. (2013). Appearance-based politics: Sex-typed facial cues communicate political party affiliation. Journal of Experimental Social Psychology, 49(1), 156-160.
[2] Rule, N. O., Ambady, N. (2010). Democrats and Republicans can be differentiated from their faces. PloS one, 5(1), e8733.
[3] Cohen, R., Ruths, D. (2013, June). Classifying political orientation on Twitter: It's not easy!. In ICWSM.
[4] Conover, M. D., GonÃğalves, B., Ratkiewicz, J., Flammini, A., Menczer, F. (2011, October). Predicting the political alignment of twitter users. In Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on (pp. 192-199). IEEE.
[5] Iyyer, M., Enns, P., Boyd-Graber, J., Resnik, P. (2014). Political ideology detection using recursive neural networks. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 1113-1122).
[6] Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions. In EMNLP
[7] Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In EMNLP
[8] Andrew D. Martin and Kevin M. Quinn. 2002. "Dynamic Ideal Point Estimation via Markov Chain Monte Carlo for the U.S. Supreme Court, 1953-1999." Political Analysis. 10:134-153
[9] Bonica, Adam. 2016. Database on Ideology, Money in Politics, and Elections: Public version 2.0 [Computer file]. Stanford, CA: Stanford University Libraries. <https://data.stanford.edu/dime>
[10] George Lakoff. 2002. Moral Politics: How Liberals and Conservatives Think, Second Edition. University of Chicago Press.