Predicting Sentencing Disparity Using Legal Opinions

Taurean Parker
Joseph Song
Daitao Xing

DS-GA 1003: Machine Learning
Final Project
New York University

Abstract

Studies in law and economics have shown judge bias may lead to disparities in sentencing. Our project looks to expand on this research using natural language processing techniques. We see if judge opinions and the text features derived from this corpus can help explain the disparity in sentencing for federal court cases. Our findings are mixed but broadly speaking suggest that judge opinions do help in predicting disparity in sentencing. For example, running an OLS regression using judge biographical features with TF-IDF text features can help reduce the mean squared error by approximately 24 percent relative to a naive prediction (the mean of harshness of the training data) on the test dataset. We also build a hierarchical attention network which can embed documents into vectors and get a meaningful representation feature. We end this paper considering potential issues and ways to adjust for these issues which may improve model performance.

Supervised by
Professors Elliot Ash & Daniel Chen

May 15, 2017

1 Question

Studies in law and economics have shown judge bias may lead to disparities in sentencing. Recent analysis of federal court cases gathered by the Transactional Records Access Clearinghouse (TRAC) from FY 2007 to FY 2011 shows wide variation in sentencing conditioned on the crime and district [NYT02]. One study finds that after controlling for criminological, demographic, and socioeconomic variables, blacks, males, uneducated and low income groups receive longer sentences than their peers [Mus01]. Our project looks to expand on this research using natural language processing techniques to see if judge opinions and the text features derived from this corpus can help explain and predict the disparity in sentencing.

2 Dataset

We use federal sentencing data which comes from the US Sentencing Commission. The publicly available dataset does not include judge information (e.g. names, demographic characteristics, etc). Using FOIA requests, the dataset has been merged with judge information. This allows us to match the sentencing data with the judge biographical information and also the judge opinions corpus.

We have two separate datasets we consider in our project: dataset 1 and dataset 2. Each dataset has it's pros and cons. First, dataset 1 has various defendant-related features such as the type of crime committed and sex and race of the defendant (we discuss why this is important in constructing disparity measures below) but the number of cases are limited to approximately 230,000 and span from 2002-2011. Dataset 2 has limited defendant variables however the dataset contains over 930,000 cases and runs from 1971-2012 (Note that although the data spans a much wider time period, there's only a handful of cases between 1971 and 1990.).

3 Data Processing and Construction of the Disparity Measures

3.1 Construction of the disparity variables

We first consider disparity in sentencing by the judge. The first disparity measure we construct is a "harshness" variable. We use the *senttot* variable from the USSC dataset to construct our harshness features. USSC defines *senttot* as the prison term (in months) decided on by the judge. Note that this excludes any other sentencing (i.e. probation, house arrest, etc) handed down by the judge. We construct the harshness variable following Lim et. al. (2015) [ea15]:

$$harshness_{i,j,k,t} = \frac{senttot_{i,j,k,t} - min}{max - min}$$

where min and max are the statutory min and max from the USSC data set and

$$i = \text{case},\ j = \text{judge},\ k = \text{district},\ t = \text{year}$$

To control for potential unobserved heterogeneity, we demean the harshness variable by district-crime-year which we define as $z_{i,j,k,t}$:

$$z_{i,j,k,t} = harshness_{i,j,k,t} - \overline{harshness_{c,k,t}}$$

$$c = \text{type of crime}$$

(Note that this is where the defendant characteristic becomes useful since we can control the type of crime that was committed.)

Since we are considering the "harshness" for each judge, we aggregate up the dataset by taking the average of the demeaned harshness measure by judge-year where:

$$\bar{z}_{j,t} = \frac{\sum_{i=1}^{m} z_{i,j,k,t}}{m}$$

Where m is the number of cases that judge j saw in year t. We consider a harshness measures for each year separately for each judge under the assumption that it is possible that the judges' bias may evolve over time. After aggregating up to the judge-year level, our data set shrinks to approximately 4800 observations.

Note that Dataset 2 doesn't have statutory min and max variables so we only have the sentencing variable *prisonsentencemonths* (prison sentence in month). Moreover, we do not have the crime that was committed by the defendant. Therefore, we standardize the data by taking the log of the sentencing variable and follow similar construction from above but just demean by district-year to control for any unobserved heterogeneity at the district-year level:

$$harshness_{i,j,k,t} = \log prisonsentencemonths_{i,j,k,t}$$

where

$$i = \text{case},\ j = \text{judge},\ k = \text{district},\ t = \text{year}$$

Then we demean the harshness variable by district-year which we define as $z_{i,j,k,t}$:

$$z_{i,j,k,t} = harshness_{i,j,k,t} - \overline{harshness_{k,t}}$$

then we average the demeaned harshness by judge-year where:

$$\bar{z}_{j,t} = \frac{\sum_{i=1}^{m} z_{i,j,k,t}}{m}$$

Where m is the number of cases that judge j saw in year t. After aggregation up to the judge-year level, we are left with approximately 14,600 observations.

3.2 Construction of the disparity variable

Additionally for dataset 1 we consider whether or not the there is heterogeneous treatment by the judge based on defendant characteristics (e.g. race, sex).

We first demean the harshness variable for judge-crime-year:

$$z_{c,i,j,t} = harshness_{c,i,j,t} - \overline{harshness_{c,j,t}}$$

where

$$c = \text{type of crime } i = \text{case},\ j = \text{judge},\ t = \text{year}$$

Once, we demean the harshness variable, we take the average harshness for group l (e.g. race or sex) by judge-year.

$$\bar{z}_{j,l,t} = \frac{\sum_{i=1}^{m} z_{i,j,l,t}}{m}$$

where l is the different groups. For example, race $i = black, notblack$. For sex $i = male, female$. m is the number of observations for each group for judge year.
Then we calculate the disparity measure as:

$$disparity_{j,t} = \bar{z}_{j,l',t} - \bar{z}_{j,l'',t}$$

Where l' and l'' are the different groups. Note that after dropping any missing values due to the possibility that the judge only saw cases of one race or one gender, our dataset shrinks to approximately 3500 for the sex disparity measure and to approximately 4200 for the race disparity measure.

3.3 Non-text features

We consider various defendant and judge characteristics that may help predict disparity in sentencing. For judge characteristics, these are:

- Christian (y/n)
- Rated highly by the ABA (y/n)
- Black (y/n)
- Appointed by a Democrat President (y/n)
- Age/Years on Bench
- Gender (M/F)

For Dataset 1, we also consider defendant characteristics:

- Did the case go to trial (y/n)
- Number of counts brought against the defendant
- Race
- Gender

3.4 Construction of the text data

Using the judge name we are then able to match the sentencing data to the opinion corpus. From the corpus, we construct various text features using different kinds of NLP techniques. The methods we used in this project includes n-grams, TF-IDF and topic models.

We first count all frequency words in text data. Then build bi-grams to five-grams features. For each n-grams feature, the 500k most frequent words from the datasets are selected. As the distribution of words are quiet different, TF-IDF will scale the counts into a more reasonable features. Topic model will compress the n-grams features into n-dimension vectors.

Besides those classic methods, we also implement a hierarchical attention model for harshness regression. RNN can make use of more context information than n-grams. In addition, attention mechanism will give different weights to words and sentences in different context and show which words contributes more to the harshness. This result will in turn guide a more efficient n-grams construction.

4 Models

We run a gamut of models to analyze the predictiveness of the text features ranging from a standard OLS regression to a more complex hierarchical attention model. A list and a brief explanation are below.

4.1 Standard models

- Ordinary Least Squares
- Partial Least Squares (We consider 5 components)
- Random Forest
- Elastic Net ($\alpha = 0.1$ and $l1$ ratio $= 0.7$)

Elastic net parameters correspond to the sk-learn elastic net function parameters.

4.2 Hierarchical Attention Model

There are limits to what can be done with n-grams. N-grams depends on searching through a large dictionary and essentially doing template matching which only considers limited information about the context. The position information included in n-grams is up to N. When we set N larger and larger, there will be more and more tokens combination in dictionary. On the other hand, the frequency of each N-grams become sparse. Thus make it much harder to predict label. Another drawback when using N-grams features is that training a model becomes really time-costly when the N-grams dictionary is huge.

In this context, we propose using a hierarchical attention network. Hierarchical attention network can capture insights in document structure. As shown in Figure 1, Hierarchical attention network includes five layers [ea16]. The first layer is a word encoder. Given a word in one sentence, we first embed the word through an embedding matrix W. This matrix is initialized through a word2vec model trained by gensim. Then we use a bidirectional GRU to get annotations of words by summarizing the information from both direction of the words. In the second layer, the word attention layer will extract words that are more important to the meaning of the sentence and aggregate the representation of those informative words to form a sentence vector. The third layer contains the sentence encoder and the forth layer works similarly to the former two layers. After the sentence attention layer, we will get a length fixed document vector representation which will be feed into a regression layer to predict harshness.

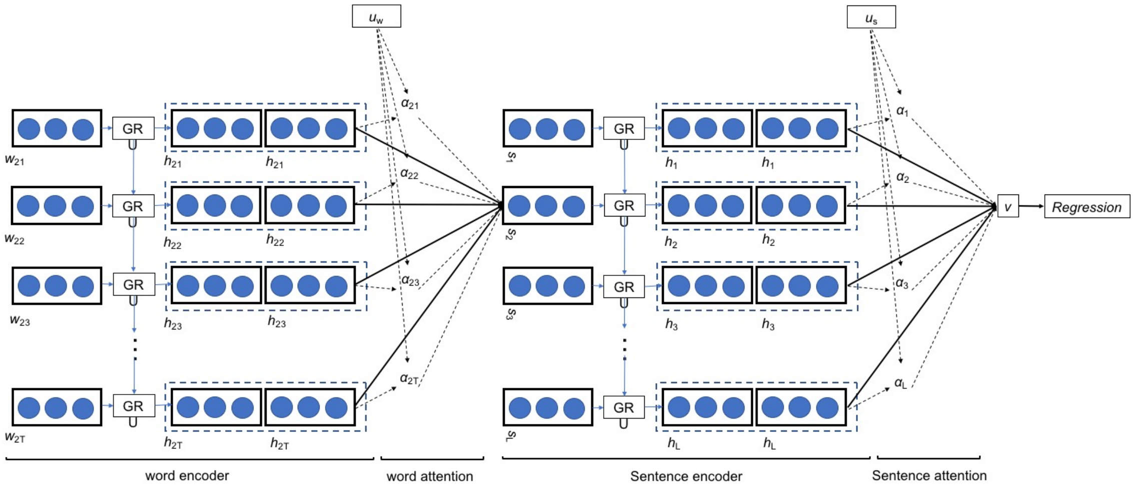

Figure 1: Hierarchical Attention Model

The base cell used in this network are called GRU cell. The GRU cell uses a gating mechanism to track the state of sequences without using separated memory cells. After GRU cells, the words in sentences will be embedded into vectors. Not all words contributes equally to the representation of the sentence meaning. Attention mechanism will extract such importance words by give each word different weights. After sentences are transformed into vectors, we can then apply similar method to transform documents into vectors.

To better predict the judges' harshness on different crime type, we represent harshness as a vector rather than a label. Given a judge in year t, we have a harshness vector:

$$Z_{j,k,t} = [z_{c1,j,k,t}, z_{c2,j,k,t}, \ldots, z_{c22,j,k,t}]$$

where c_i is the type of crime.

For most judges, they only adjudicated in some specific type of cases which means they will never have harshness on other types of crimes in the dataset. To get rid of this influence, we only consider loss on the columns which are not zeros:

$$loss = \frac{1}{m}\sum_{i=1}^{m}(y_{pred}^{i} - y_{true}^{i})^2$$

where m is the number nonzero harshness in the vectors.

5 Results

5.1 Main Results

For our analysis we split the dataset 80/20 for training and testing, respectively. Once we've trained our model, we compare the predicted disparity measures from the test dataset to the actual disparity measures and calculate the mean squared error. Our baseline is a "naive prediction" where for each test observation we predict the mean of the disparity measures from the training data. Any improvement from the "naive prediction" suggest the model has some predictive lift. Our primary results are in tables 1 through 4.

When we try to predict the "harshness" disparity measures (Table 1 & 4), our model with no text features which only includes judge and (when available) defendant characteristics only marginally improves prediction for ordinary least square and partial least squares. As for random forest and elastic net, we see no lift in prediction accuracy. In fact, random forest actually performs worst than the naive prediction (we'll discuss the reasons why we didn't get more lift from the no text features model below in this section and in Section 6). When we subsequently add text features into our model, the results are mixed. For the harshness variable (Table 1 & 4), we see lift in accuracy in the ordinary least squares, random forest and elastic net models.

For the disparity measures, performance improvement was quite modest or little changed (Table 2 & 3). We see some predictive lift when adding text features in to the OLS and elastic net models for gender disparity measure. However, there's little improvement when predicting race disparity measures. In fact, performance deteriorates slightly. Random forest models perform poorly for both disparity measures.

For partial least squares, we did not come up with any results as our models did not solve when using 5 components (we'll discuss further in section 6 which may help us get results).

Table 1: Mean Squared Error (MSE): Dataset 1 - Predicting Harshness

	OLS	PLS	Random Forest	Elastic Net
Naïve Prediction	0.0038	0.0038	0.0038	0.0038
No Text Features	0.0037	0.0037	0.0044	0.0038
BOW + TF-IDF	0.0029	–	0.0033	0.0028
Bigrams	0.0029	–	0.0032	0.0028
Trigrams	0.0028	–	0.0032	0.0028
Fourgrams	0.0029	–	0.0030	0.0028
Fivegrams	0.0029	–	0.0031	0.0028

Table 2: MSE - Dataset 1 Disparity Measure: Sex

	OLS	PLS	Random Forest	Elastic Net
Naïve Prediction	0.0070	0.0070	0.0070	0.0070
No Text Features	0.0069	0.0069	0.0088	0.0070
BOW + TF-IDF	0.0069	–	0.0071	0.0066
Bigrams	0.0066	–	0.0079	0.0066
Trigrams	0.0066	–	0.0074	0.0066
Fourgrams	0.0065	–	0.0072	0.0066
Fivegrams	0.0066	–	0.0076	0.0066

Table 3: MSE - Dataset 1 Disparity Measure: Race

	OLS	PLS	Random Forest	Elastic Net
Naïve Prediction	0.0100	0.0100	0.0100	0.0100
No Text Features	0.0992	0.0991	0.0135	0.0100
BOW + TF-IDF	0.0103	–	0.0108	0.0101
Bigrams	0.0102	–	0.0111	0.0101
Trigrams	0.0101	–	0.0106	0.0101
Fourgrams	0.0101	–	0.0109	0.0101
Fivegrams	0.0101	–	0.0108	0.0101

Table 4: MSE - Dataset 2 Predicting Harshness

	OLS	PLS	Random Forest	Elastic Net
Naïve Prediction	0.72	0.72	0.72	0.72
No Text Features	0.69	0.69	0.85	0.72
BOW + TF-IDF	0.68	–	0.74	0.68
Bigrams	0.68	–	0.75	0.68

As we mentioned in the data construction section, we residualized (or demeaned) the dependent variable at either the district-crime-year level or judge-crime-year level depending on the disparity measure (harshness vs gender and race disparity measures). Figure 2 shows the distribution of crimes charged to the defendant in dataset 1. Given that the distribution of the crimes is quite uneven, it's possible that when we residualize, some "cells" (e.g. a specific district i, crime c and year t) had only a few observations making the harshness measure imprecise and potentially skewed by outliers or because all the values were the same suggest there is no disparity or the same harshness. (We consider ways we could have alleviate this issue in Section 6).

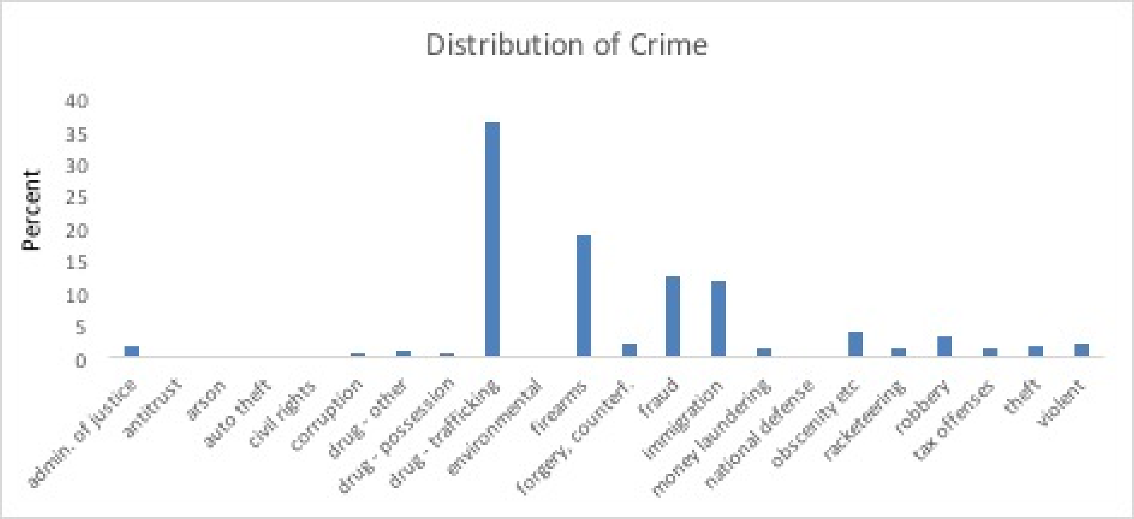

Figure 2: Distribution of Primary Crime Committed

One way we resolve this issue is by only residualizing on district-year or judge-year levels and use the type of crime and a feature to predict the disparity measures. Tables 5-7 shows these results. We find marked improvement across all three disparity measures when going from naive prediction to no text features model to adding text features. This suggests that the crime feature holds significant predictive value in how predicting disparity in sentencing.

Table 5: MSE - Dataset 1 (Only Residualize on District Year) Predicting Harshness

	OLS	PLS	Random Forest	Elastic Net
Naïve Prediction	0.0065	0.0065	0.0065	0.0065
No Text Features	0.0053	0.0053	0.0054	
BOW + TFIDF	0.0039	–	0.0041	0.0047
Bigrams	0.0038	–	0.0044	0.0047
Trigrams	0.0039	–	0.0043	0.0047
Fourgrams	0.0039	–	0.0043	0.0047
Fivegrams	0.0039	–	0.0042	0.0047

Table 6: MSE - Dataset 1 (Only Residualize on Judge Year) Disparity Measure: Sex

	OLS	PLS	Random Forest	Elastic net
Naïve Prediction	0.0199	0.0199	0.0199	0.0199
No Text Features	0.0149	0.0154	0.0168	0.0199
BOW + TFIDF	0.0152	–	0.0177	0.0195
Bigrams	0.0148	–	0.0167	0.0195
Trigrams	0.0147	–	0.0159	0.0195
Fourgrams	0.0148	–	0.0159	0.0195
Fivegrams	0.0150	–	0.0164	0.0195

Table 7: MSE - Dataset 1 (Only Residualize on Judge Year) Disparity Measure: Race

	OLS	PLS	Random Forest	Elastic net
Naïve Prediction	0.0233	0.0233	0.0233	0.0233
No Text Features	0.0187	0.0189	0.0213	
BOW + TFIDF	0.0166	–	0.0184	0.0211
Bigrams	0.0166	–	0.0180	0.0211
Trigrams	0.0165	–	0.0186	0.0211
Fourgrams	0.0165	–	0.0179	0.0211
Fivegrams	0.0166	–	0.0181	0.0211

5.2 Random Forest: Word Clouds and Relative Importance

The word cloud figures below are based on the relative importance from the random forest regression algorithm (Figures 3-8). Words that appear larger on the plots are features that the random forest measure deemed important in measuring the sentencing disparity variable. However, to determine the relationship between the text features and the sentencing disparity variable, we ran univariate regression analysis. The n-grams shown in tables 8-11 are based on whether or not the feature are positively or negatively affect the disparity variable based on the relative importance of the random forest alorigthm. All the features that were selected are statistically significant using the F-regression statistic in sklearn.

Bigram features do not provide clear interpretable results. That is, the phrases that are considered "important" do not seem to have much relation to language that would be considered bias. Using bigrams would require additional information to determine the relationship between words and the harshness sentencing variable. For example, we could query court opinions based on the following bigrams and determine a similarity measure. In contrast, trigrams performed slightly better in terms of interpretability. Our results show that court opinions that contain the words “arkansa western division” or “Mississippi” are positive for racial disparity. In both the bigrams and trigrams tables, court opinions with word “Mississippi” were positive for racial disparity, meaning harsher sentencing for blacks relative to non-blacks. This could be picking up on the racial issues and implicit biases that may still be prevalent in the South.

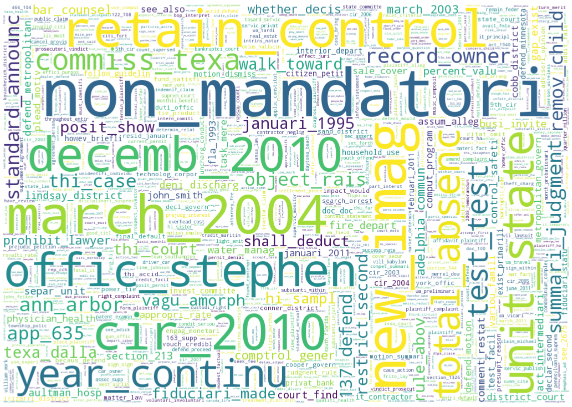

Figure 3: Bigram Word Cloud: Dataset 1 - Predicting Harshness

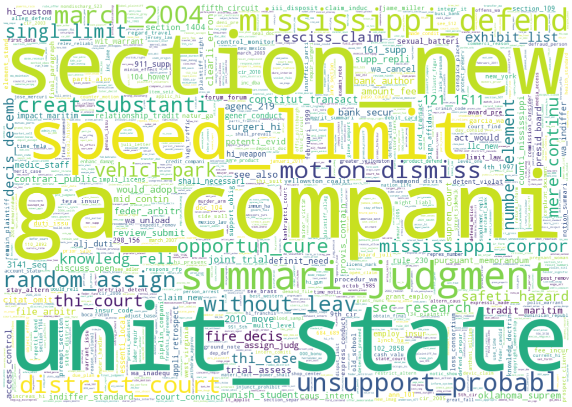

Figure 4: Bigram Word Cloud: Dataset 1 - Predicting Race Disparity

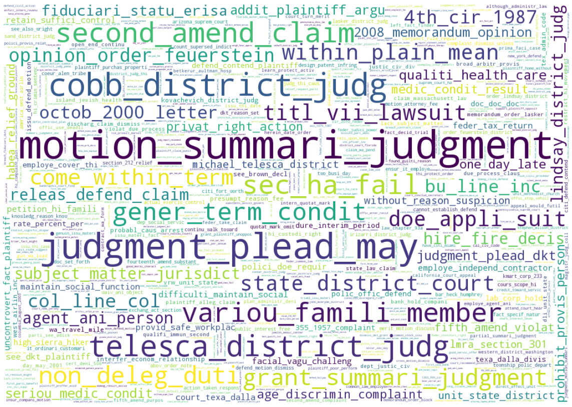

Figure 5: trigram Word Cloud: Dataset 1 - Predicting sex Disparity

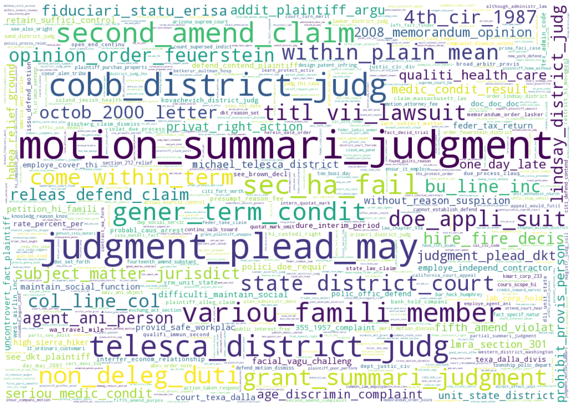

Figure 6: Trigram Word Cloud: Dataset 1 - Predicting Harshness

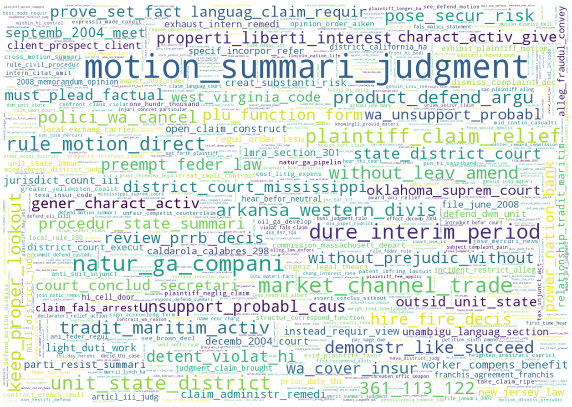

Figure 7: Trigram Word Cloud: Dataset 1 - Predicting Race Disparity

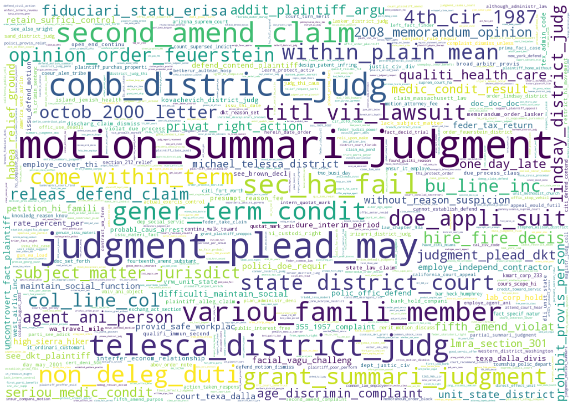

Figure 8: Trigram Word Cloud: Dataset 1 - Predicting sex Disparity

Table 8: Bigrams that are Positive for Disparity Measures (From Univariate Regression)

Harshness	Disparity - Race	Disparity - Sex
new_imag	ga_compani	june_2001
march_2004	section_new	mbna_bank
retain_control	speed_limit	aircraft_oper
unit_state	mississippi_defend	depart_taxat
year_continu	creat_substanti	stop_limit
total_absenc	singl_limit	regularli_schedul
commiss_texa	mississippi_corpor	limit_coverag
test_test	mere_continu	feder_permit
record_owner	number_element	use_defens
summari_judgment	knowledg_reli	mental_capac
ann_arbor	safeti_hazard	test_period
standard_announc	decis_decemb	use_provis
thi_case	sec_research	state_oregon
app_635	fire_decis	undu_influenc
januari_1995	121_1511	mccarran_ferguson
posit_show	review_submit	defend_constitut
vagu_amorph	surgeri_hi	resolut_provis
thi_court	oklahoma_suprem	jersey_superior
rule_abov	2010_move	distribut_busi
shall_deduct	tradit_maritim	trust_deed

Table 9: Bigrams that are Negative for Disparity Measures (From Univariate Regression)

Harshness	Disparity - Race	Disparity - Sex
non_mandatori	unit_state	deni_intervenor
cir_2010	summari_judgment	er_state
decemb_2010	unsupport_probabl	unit_state
offic_stephen	march_2004	summari_judgment
object_rais	motion_dismiss	point_posit
remov_child	district_court	busi_enterpris
137_defend	random_assign	notic_sale
texa_dalla	without_leav	2003_iep
restrict_second	opportun_cure	doc_124
walk_toward	vehicl_park	prepar_time
prohibit_lawyer	thi_court	constitut_occurr
adelphia_commun	thi_case	proof_applic
comptrol_gener	resciss_claim	close_end
march_2003	exhibit_list	district_court
lindsay_district	constitut_transact	actual_expend
deni_discharg	would_adopt	worker_paid
separ_unit	punish_student	hear_initi
act_intermediari	feder_arbitr	pleasant_valley
gap_exist	161_supp	statement_statement
water_manag	contrari_public	thi_case

Table 10: Trigrams that are Positive for Disparity Measures (From Univariate Regression)

Harshness	Disparity - Race	Disparity - Sex
motion_summari_judgment	natur_ga_compani	alj_determin_claimant
judgment_plead_may	market_channel_trade	disput_resolut_provis
cobb_district_judg	dure_interim_period	larsen_civil_right
telesca_district_judg	rule_motion_direct	bodili_injuri_coverag
sec_ha_fail	arkansa_western_divis	work_daili_basi
variou_famili_member	keep_proper_lookout	fair_busi_practic
gener_term_condit	361_113_122	marin_fisheri_serv
non_deleg_duti	tradit_maritim_activ	658_4th_cir
grant_summari_judgment	pose_secur_risk	profit_share_plan
state_district_court	polici_wa_cancel	complaint_fail_suffici
titl_vii_lawsuit	demonstr_like_succeed	jurisdict_defend_alleg
within_plain_mean	hire_fire_decis	strong_infer_deliber
doe_appli_suit	preempt_feder_law	prairi_band_potawatomi
come_within_term	procedur_state_summari	decis_plaintiff_disabl
opinion_order_feuerstein	wa_cover_insur	decis_deni_coverag
col_line_col	plu_function_form	hi_cell_door
subject_matter_jurisdict	review_prrb_decis	violat_ani_requir
agent_ani_person	district_court_mississippi	prudenti_in_america
releas_defend_claim	gener_charact_activ	hog_mkt_inc
hire_fire_decis	charact_activ_give	februari_2011_plaintiff

Table 11: Trigrams that are Negative for Disparity Measures (From Univariate Regression)

Harshness	Disparity - Race	Disparity - Sex
second_amend_claim	motion_summari_judgment	182_fed_appx
4th_cir_1987	plaintiff_claim_relief	motion_summari_judgment
bu_line_inc	unit_state_district	sentenc_thi_section
octob_2000_letter	product_defend_argu	pay_minimum_wage
lindsay_district_judg	without_leav_amend	offic_reason_suspicion
judgment_plead_dkt	properti_liberti_interest	dismiss_qui_tam
one_day_late	unsupport_probabl_caus	gener_maritim_law
age_discrimin_complaint	detent_violat_hi	unit_state_district
prohibit_provis_person	prove_set_fact	malpractic_claim_defend
without_reason_suspicion	state_district_court	ei_arbitrari_caprici
feder_tax_return	without_prejudic_without	585_106_1348
court_texa_dalla	must_plead_factual	fraud_neglig_misrepresent
petition_hi_famili	wa_unsupport_probabl	set_asid_program
see_dkt_plaintiff	outsid_unit_state	fraud_breach_fiduciari
uncontrovert_fact_plaintiff	worker_compens_benefit	limit_mean_plu
high_sierra_hiker	client_prospect_client	hi_motion_attorney
texa_dalla_divis	jurisdict_count_iii	see_fed_evid
wa_travel_mile	claim_fals_arrest	llc_wa_form
day_may_2001	decemb_2004_court	must_plead_enough
bank_hold_compani	file_june_2008	accept_respons_action

5.3 Result of Hierarchical Attention Model

In our experiment, we set the word embedding dimension to 100 and GRU dimension to 50. The word context vectors have a dimension of 100, initialized by word2vec model trained by gensim package.

We use adam (a kind of SGD version) as our optimizer and batch size to 16. As shown in Figure 9, the mean squared error and mean absolute error converge after 20 epochs.

Due to time constraints, we run the hierarchical attention model on just dataset 1. The squared loss on test data is 0.0049, which exceeds the performance of the naive prediction and baseline (no text features) model. It also exceeds the performance of pure n-grams model.

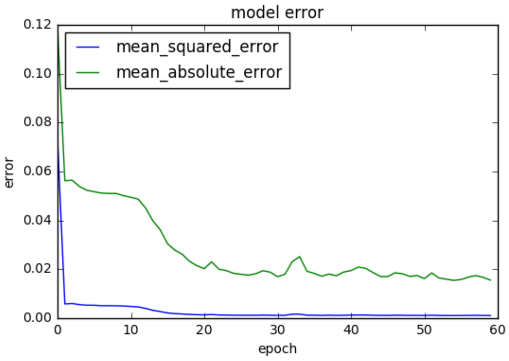

Figure 9: Mean squared loss and mean absolute loss on test data.

Hierarchical attention allows for an efficient way to convert text documents into vectors. We extract a vector representation from the network and then combine it with judge-level and defendant-level features to get a new feature dataset. As show in figure 10, these features capture more useful information about judges' harshness in sentencing.

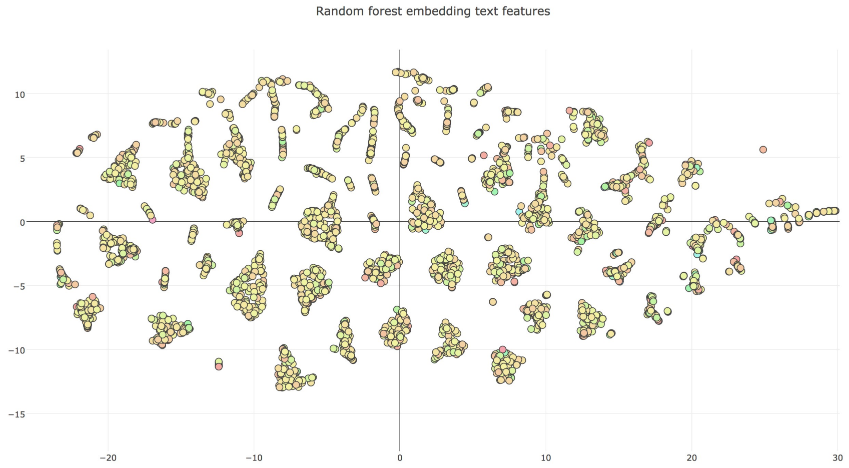

Figure 10:   Random Forest embedding text features

6 Discussion and Next Steps

Our analysis shows to a certain degree that judge opinion corpus may hold valuable information about a judges' bias and help predict disparity in sentencing. But we see various ways we can improve our analysis.

1. **Partial least squares**: As we mentioned in our results section, our models with text features did not converge. One issue may have been that we had included all n-grams from the corpus in our analysis. One way we could reduce the number of n-grams in our analysis is to run univariate regression on each n-gram present on the disparity measure and only keep the one that are statistically significant. Another option would be to remove the most common and least common n-grams from the analysis. All these options could help reduce the number of n-grams in our models and could help the PLS model converge.

2. **Crime feature**: We showed that we got better results when we residualized just on judge-year or district-year level and used crime as an feature to predict disparity. It's possible that when we residualized also on crime, we got some cells that had only a few observations, creating imprecise disparity measures. Given more time, we could have perhaps aggregated up the different crime categories to have a more even distribution which may helped produce more reliable disparity measures.

3. **Evaluate our models on actual text data**: Another way we could show robustness of our models would have been to run the models on current real life text data. One possibility would be to run a corpus of speeches or writings of prominent judges or attorney generals who we have a relatively good understanding of their stance on sentencing. For example, we would hope to see that writings by Jeff Sessions (who is considered "tough on crime") would "predict" a higher harshness score while Eric Holder (who as attorney general gave prosecutors more lee-way in charging criminals) would "predict" a lower harshness score.

4. **Other text features**: There is an extensive library of natural language processing techniques to transform a text corpus into features. Text features could be further broken down into different parts of speech, proportion of negative and positive words, or deep learning methods such as Doc2Vec, Sent2Vec, or Glove2Vec. Hierarchical attention model is proven to work on regression problems and we have shown that we get good results. However, this network only uses the features extracted from texts. It will give better performance if we can combine the network embedding vector and judge-level features and defendant-level features and build an end to end architecture.

References

[ea15] Lim et. al. The Judge, the Politician, and the Press: Newspaper Coverage and Criminal Sentencing Across Electoral systems. *American Economic Journal*, 7(4), 2015.

[ea16] Yang et. al. Hierarchical Attention Networks for Document Classification. *Proceedings of NAACL-HLT*, 2016.

[Mus01] David Mustard. Racial, Ethnic, and Gender Disparities in Sentencing: Evidence from the US Federal Courts. *Journal of Law and Economics*, XLIV, 2001.

[NYT02] Wide Sentencing Disparity Found Among US Judges. *New York Times*, March 5, 2002.