

Judge Vectors: Spatial Representations of the Law using Document Embeddings

Elliott Ash and Daniel L. Chen*

April 5, 2018

Abstract

Recent work in natural language processing represents language objects (words and documents) as dense vectors that encode the relations between those objects. This paper explores the application of these methods to legal language, with the goal of understanding judicial reasoning and the relations between judges. In an application to federal appellate courts, we show that these vectors encode information that distinguishes courts, time, and legal topics. The vectors do not reveal spatial distinctions in terms of political party or law school attended, but they do highlight generational differences across judges. We conclude the paper by outlining a range of promising future applications of these methods.

1 Introduction

An old literature in law and economics models law as a case space, where the facts are located in space, and the law separates the fact space into “liable” and “not liable” or “guilty” and “not guilty.”¹ These models give us some intuition into the legal reasoning process. But they have been somewhat limited empirically because it has been infeasible to measure the legal case space.

Meanwhile, recent work in computational linguistics has made breakthroughs in vector representations of language (Blei, 2012; Mikolov et al., 2013; Jurafsky and Martin, 2014).

*Elliott Ash, Assistant Professor of Economics, University of Warwick, e.ash@warwick.ac.uk. Daniel L. Chen, Professor of Economics, University of Toulouse, daniel.chen@iast.fr. We thank Brenton Arnaboldi, David Cai, Matthew Willian, and Lihan Yao for helpful research assistance.

¹Cameron and Kornhauser (2017) provide a recent review of this literature.

For example, the success of Google’s Word2Vec algorithm is that it “learns” the conceptual relations between words; a trained model can produce synonyms, antonyms, and analogies for any given word (Mikolov et al., 2013; Levy et al., 2015). These “word embeddings,” as the word vectors have come to be called, serve well as features in down-stream prediction tasks by encoding more information in relatively rare word features. More recently, “document embeddings” have built upon the success of word embeddings and represented words and documents in a joint geometric space (Le and Mikolov, 2014). Like word embeddings, these document embeddings have advantages in terms of interpretability and serve well in prediction and classification tasks.

Law is embedded in language. This paper asks what might be gained by applying the idea of word and document vectors to the law. The idea is to vectorize judicial rulings based on the language in those opinions. Then one can understand the relations between rulings, and between judges, using vector algebra.

In this paper, we outline recent advances in embeddings models and discuss their application in a legal context. We provide an example of this approach by constructing document embeddings for the universe of U.S. Circuit Court decisions for the years 1970 through 2010. We then produce vectors for each judge by taking the average of the document embeddings for the cases authored by the judge.

We ask, first, whether the information recovered by our model provides a meaningful signal about a judge’s legal beliefs. We look at whether the spatial relations in these embeddings encode differences between judges on different courts, between judges of different political parties, and between judges of different biographical characteristics. We find promising results from our simple document-embeddings approach.

In the concluding section we outline a range of potential future applications for the use of embeddings models in computational analysis of law. First, one could use structured embeddings to explicitly model the relations between judges, between courts, or over time. Second, one could create citation embeddings to identify similar cases based on how often they are cited together. Third, one could use embeddings to understand differences across judges in sentiment toward policies or social groups. Fourth, one could construct judge embeddings based on their predictiveness for case outcomes, rather than just the language features.

2 Embeddings Models and the Law

A first-order problem in empirical analysis of text data is the high dimensionality of text. For computational tractability, one might ignore word order and represent a document as a frequency distribution over words. But with a large vocabulary, say 20,000 words, a document is still a high-dimensional vector.

Word embeddings came about as a dimension reduction approach in deep learning models for prediction. One represents a word as a small and dense vector (say 100 dimensions). Initially, words are randomly distributed across the vector space. But the word locations become features in the learning model, and back-propagation automatically moves the embeddings around to help the neural network perform its prediction task. In NLP settings, this typically leads to words clustering near similar words. The use of embedding layers for optimal dimension reduction has much untapped potential in empirical social science (see, e.g., Rudolph et al., 2017).

The leading implementations of word vectors are trained on NLP prediction tasks, such as predicting a word from the surrounding words in a sentence. Document embeddings, such as Le and Mikolov's (2014) paragraph vectors, use a separate embedding layer for both the word and the document to solve the prediction task. Embedding models are different from topic models (e.g. Blei, 2012) because the dimensions have a spatial interpretation, rather than a topic-share interpretation. These models have become popular because the spatial relations between the trained embeddings encode useful and meaningful information (Levy et al., 2015).

To illustrate, a word embedding can identify similar words in the vocabulary. For example, the closest word to “judge” might be “jury.” Similarly, a document embedding can identify similar cases in a corpus of decisions. For example, the closest case to *Engel v. Vitale* (1962) might be *Everson v. Board of Education* (1947). Finally, a judge embedding constructed from these documents could be used to identify similar judges in the legal system. For example, the closest judge to Antonin Scalia might be Clarence Thomas.

A more intriguing exercise is to think about analogies. A functional word embedding would be able to say that “governor” is to “state” as “mayor” is to “city,” through the vector algebra $\text{governor} - \text{state} + \text{city} = \text{mayor}$. Similarly, a document embedding could say something like “*Everson vs. Board of Education* is to *Engel v. Vitale* as *Griswold v. Connecticut* is to *Roe v. Wade*.” These cases share an analogical relation, in that the latter case is a related application of the constitutional principle articulated in the former case. In the vector math, that would be represented as $\text{Everson} - \text{Engel} + \text{Griswold} = \text{Roe}$. Finally,

a judge embedding could say something like “Scalia is to Thomas as Ginsburg is to Breyer,” in the sense that $\text{Scalia} - \text{Thomas} + \text{Breyer} = \text{Ginsburg}$.

In the case of word embeddings, the directions in the embedding space often encode semantic meaning. For example, Bolukbasi et al. (2016) show that there is a vector direction for gender in the embedding space. One can also typically isolate directions for time, singular vs plural, etc. In the legal case, we would be interested in isolating directions for legal and political concepts and distinctions. For example, there could be a direction for liberal vs conservative, or procedural vs substantive. There could be directions or clusters for originalists, or pragmatists, or economic analysis.

3 Application to Federal Appellate Courts

This section illustrates the use of document embeddings in the federal appellate courts.

3.1 Data and Documents

The analysis utilizes a corpus of all U.S. Supreme Court cases, and all U.S. Circuit Court cases, for the years 1887 through 2013. We have detailed metadata for each opinion; we mainly use the court, date, case topic, and authoring judge. For case topic, we use the 7-category “General Issue” designation coded for Donald Songer’s Court of Appeals Database. The cases are linked to biographical information on the judges obtained from the Federal Judicial Center. This includes birth date, gender, and political affiliation of appointing president.

We also have the full text of the cases. We remove HTML markup and citations. We then have each case as a list of tokens. These tokens provide the inputs for the embeddings model.

3.2 Document Vectors

The next step is to construct document vectors for each case i . The model we use is Doc2Vec (Le and Mikolov, 2014), implemented in the Python package gensim. The objective function solved by this model is to iterate over the corpus and try to predict a given word using its context (a window of neighboring words), as well as a bag-of-words representation of the whole document. The model uses an embedding layer for the context features and

the document features. Therefore the geometric location of documents encodes predictive information for the context-specific frequencies of words in the document.

We feed the case documents in random order into Doc2Vec. We used the distributed bag-of-words model over the distributed memory model, with 200 dimensions per document vector. Other parameter choices include a context window of size 10, capping the vocabulary at 100,000 words (based on document frequency), and excluding documents shorter than 40 words in length. As this chapter is an exploration and illustration, we did not substantially explore the parameter space on these margins.

3.3 Vector Centering and Aggregation

We now have a set of vectors $\vec{i}$ for each case i . Following the advice of the embeddings literature,² we normalized each vector to length one. Each case has an authoring judge j , working in court c at year t . Besides author and time, the other metadata feature is the case topic k .

For visualization and other analysis we would like to center and aggregate the document vectors in several ways. Let I_j be the set of cases authored by j . Let I_{jt} be the set of cases authored by j at year t . One could construct a vector representation for a judge using

$$\vec{j} = \frac{1}{|I_j|} \sum_{i \in I_j} \vec{i}$$

where $|\cdot|$ gives the count of the set. Similarly, the vector for judge j at year t would be given by

$$\vec{jt} = \frac{1}{|I_{jt}|} \sum_{i \in I_{jt}} \vec{i}$$

and the vector for all cases on topic k in court c during year t would be given by

$$\vec{ckt} = \frac{1}{|I_{ckt}|} \sum_{i \in I_{ckt}} \vec{i}.$$

Meanwhile, the same notation and corresponding aggregation formula could be used to construct a vector for a year, $\vec{t}$, for a court $\vec{c}$, for a topic $\vec{k}$, or for the cases in court c during a particular year t , $\vec{ct}$.

We are interested in recovering the ideological component of the judge vectors. Therefore

²See <https://www.quora.com/Should-I-do-normalization-to-word-embeddings-from-word2vec-if-I-want-to>

we explore the following steps to center the document vectors before aggregating. Represent the year-centered vector for case i as $\vec{i}_t = \vec{i} - \vec{t}_i$, where $\vec{t}_i$ corresponds to the average vector for all cases in the same year as i . Similarly, let a subscripted judge vector $\vec{j}_t$ be defined as

$$\vec{j}_t = \frac{1}{|I_j|} \sum_{i \in I_j} \vec{i}_t$$

the average for judge j of the year-centered vectors $\vec{i}_t$.

The preferred centering specification depends on the context of the analysis. We center by interacted groups, in particular. In the results below, we variously center by topic-year $\vec{kt}$, by court-year $\vec{ct}$, and by court-topic-year $\vec{ckt}$. Only after this centering step do we aggregate by judge and perform analysis of the spatial relations between vectors. The hope is that the remaining spatial variation is purged of court-specific, topic-specific, and year-specific differences in language. The remaining variation will provide a cleaner summary of the ideological differences between judges.

Here we have used the simple average. But one could imagine that it would be useful or illuminating to use a weighted average. One could weight the cases by their length (in words or sentences), for example. Alternatively, one could weight the cases by the number of citations it later received by later judges, as a proxy for importance. Finally, one could use a data-driven measure of importance or controversy, perhaps based on the lower-court opinion features.

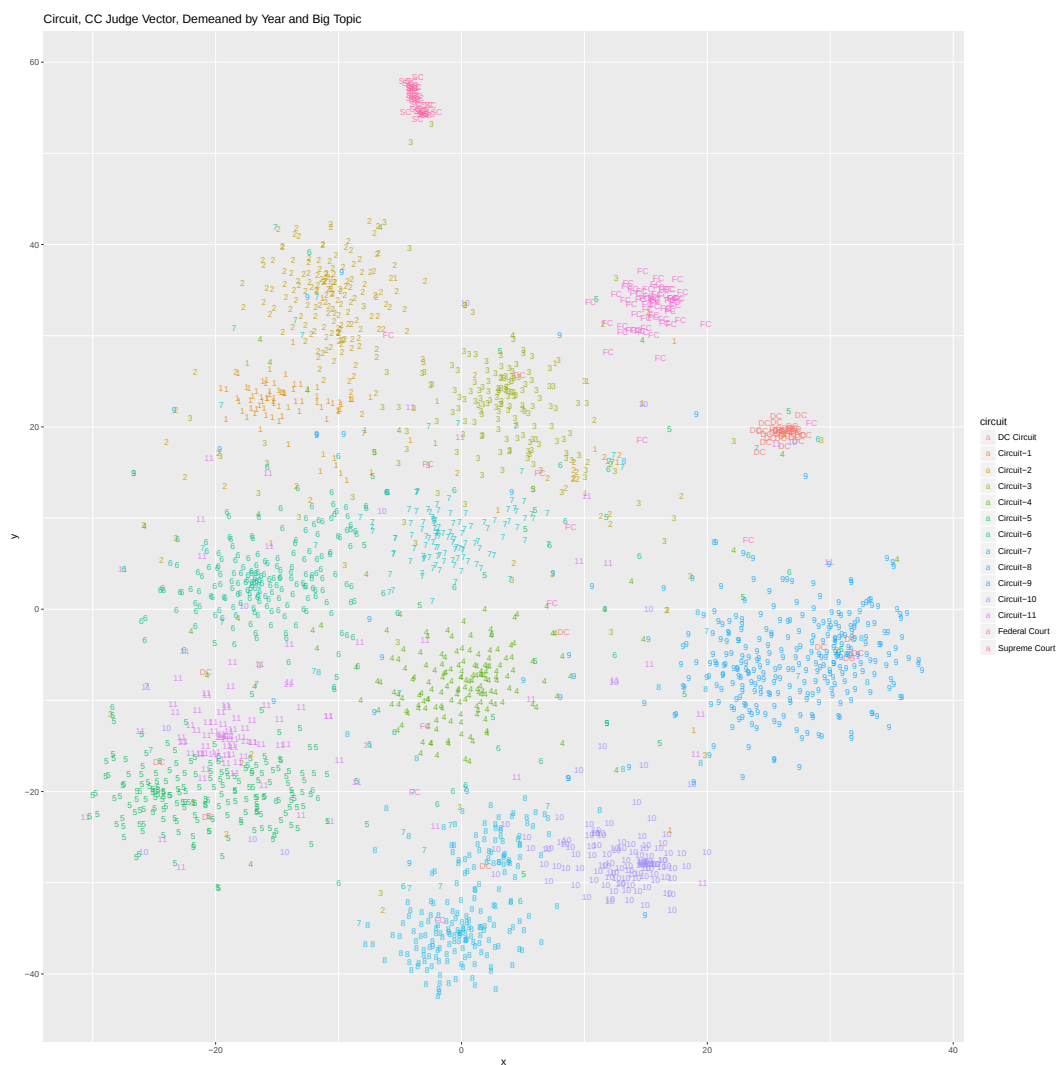
3.4 Visual Structure of Case Vectors and Judge Vectors

In this section we present a variety of visualizations to understand better the spatial relationships encoded by our case vectors and judge vectors. Our visualization method is a t-SNE plot (Maaten and Hinton, 2008), which projects the vectors down to two dimensions for visualization purposes. We use t-SNE plots, rather than PCA, because the dimension reduction algorithm is designed to project data while preserving relative distance between points. The dots represent vectors, and the colors/labels represent groupings.

We begin by exploring what institutional, temporal, and judge-level features are encoded in the vectors. For Figure 1, we de-meaned the case vectors by topic interacted with year. We then averaged by judge and plotted the judge vectors. The vectors are labeled by court. One can see that, conditional on topic and year, the document vectors separate the courts quite well.

For Figure 2, we centered on court interacted with topic. We then average by court-year

Figure 1: Centered by Topic-Year, Averaged by Judge, Labeled by Court



and plotted the court-year-level averaged vectors. We labeled and colored by the decade the case was published. One can see a steady linear development of case law across the geometric space.

For Figure 3, we centered on judge interacted with year, netting out any judge-level time-varying component of language. We then averaged and plotted by topic-year. The labels and colors are by the seven-digit general issue topic. One can see that the document embeddings discriminate topics effectively.

Next we look at whether the vectorized language in the case vectors encodes information about judge characteristics. For Figure 4, we centered on an interacted groupings for court, topic, and year. This nets out any time-varying topic and court level language variation. We then averaged by judge and plotted the judge vectors. The labels and colors are by political party – Democrat or Republican. These are randomly distributed across the graph. It appears that the language features encoded by the document embeddings are not informative about political party. This is related to the result in Ash et al. (2017) that judicial language is not very polarized relative to congressional language. One potential reason for this is that we use a bag-of-words model for text rather than a bag-of-phrases. The ideological content of the law might be represented in phrases rather than single words.

Figure 4 considers another judicial biographical feature, birth cohort. As before, we centered on court, topic, year and averaged/plotted by judge. But now, the labels and colors are by birth cohort decade (1910s through 1950s). In stark contrast to political party, there is clear segmentation across the geometric space across cohorts. Remember that this is conditioned on court-topic-year, so is not driven by time trends over the sample. The vectorized language recovers differences in the legal language used by judges from different generations.

Finally, for Figure 4, we consider law school attended as a final source of linguistic differences across judges. Conditional on court, topic, and year, we see just random distributions across the graph in terms of law school. As with political party, it seems like language or ideological differences by school do not show up in the vectors. Again, this may be due to ideological differences being represented in phrases rather than single words.

3.5 Analysis of Relations Between Judges

This section uses our vector representation of judges to produce a similarity metric between courts and judges. We adopt a measure of vector similarity that is used often on document

Figure 2: Centered by Court-Topic, Averaged by Court-Year, Labeled by Decade

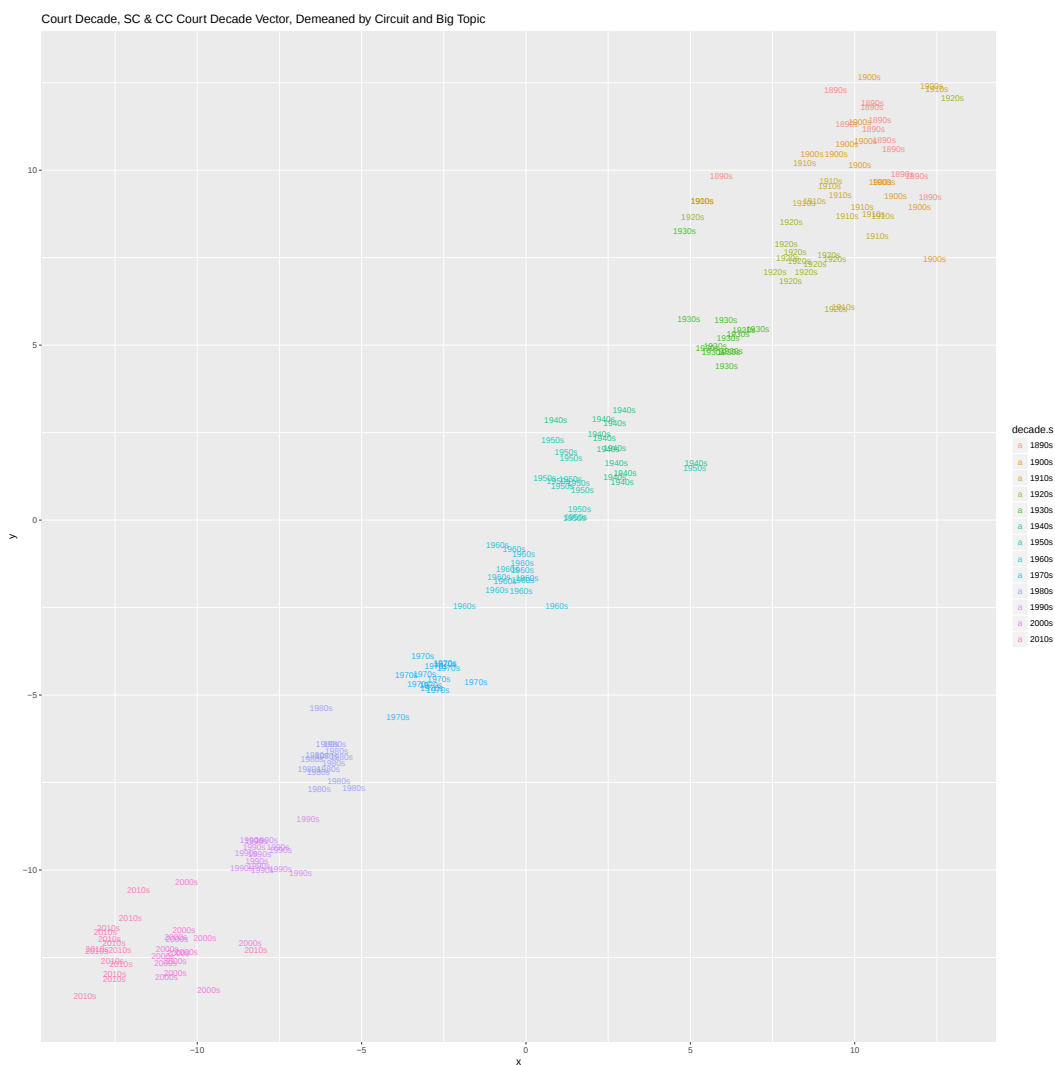


Figure 3: Centered by Judge-Year, Averaged by Topic-Year, Labeled by Topic

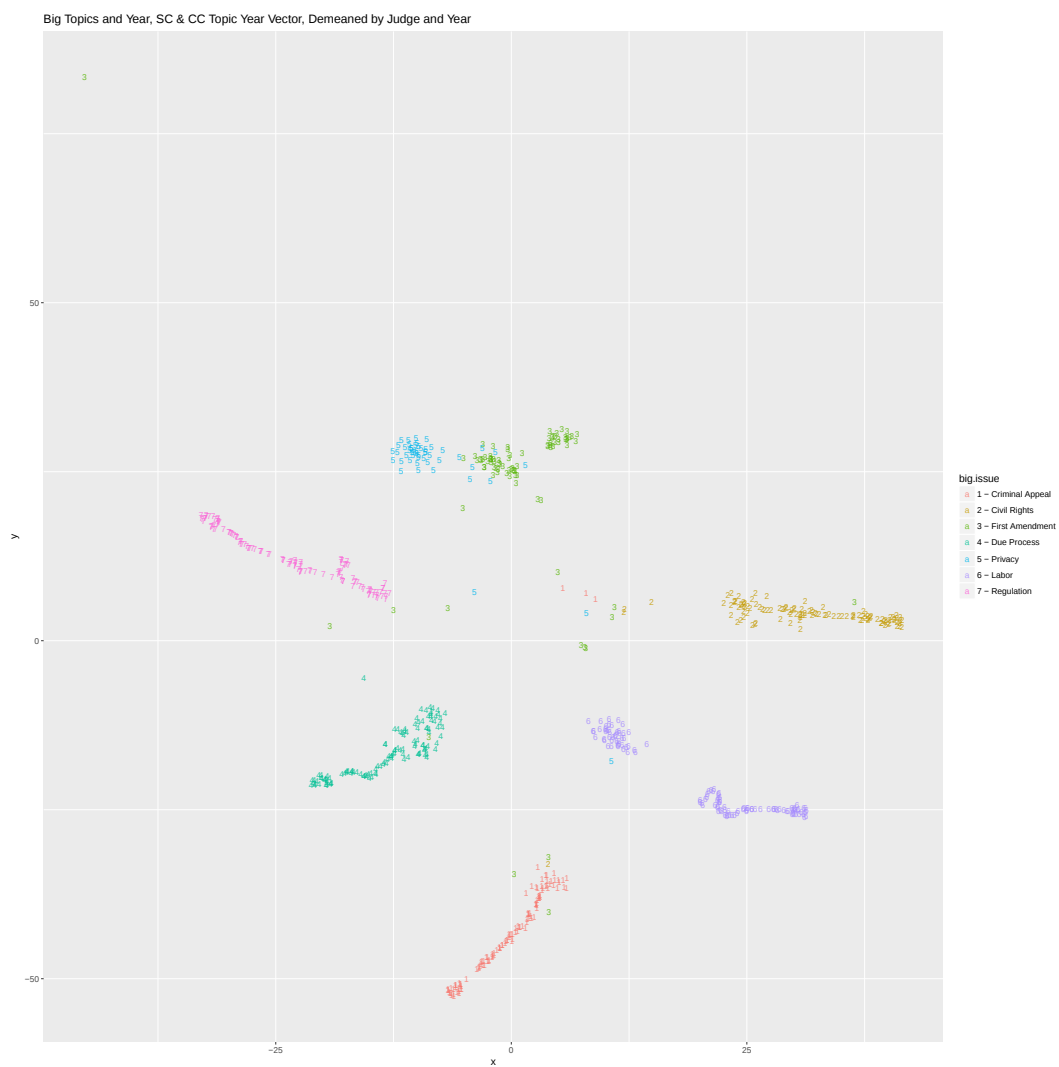


Figure 4: Centered by Court-Topic-Year, Averaged by Judge, Labeled by Political Party

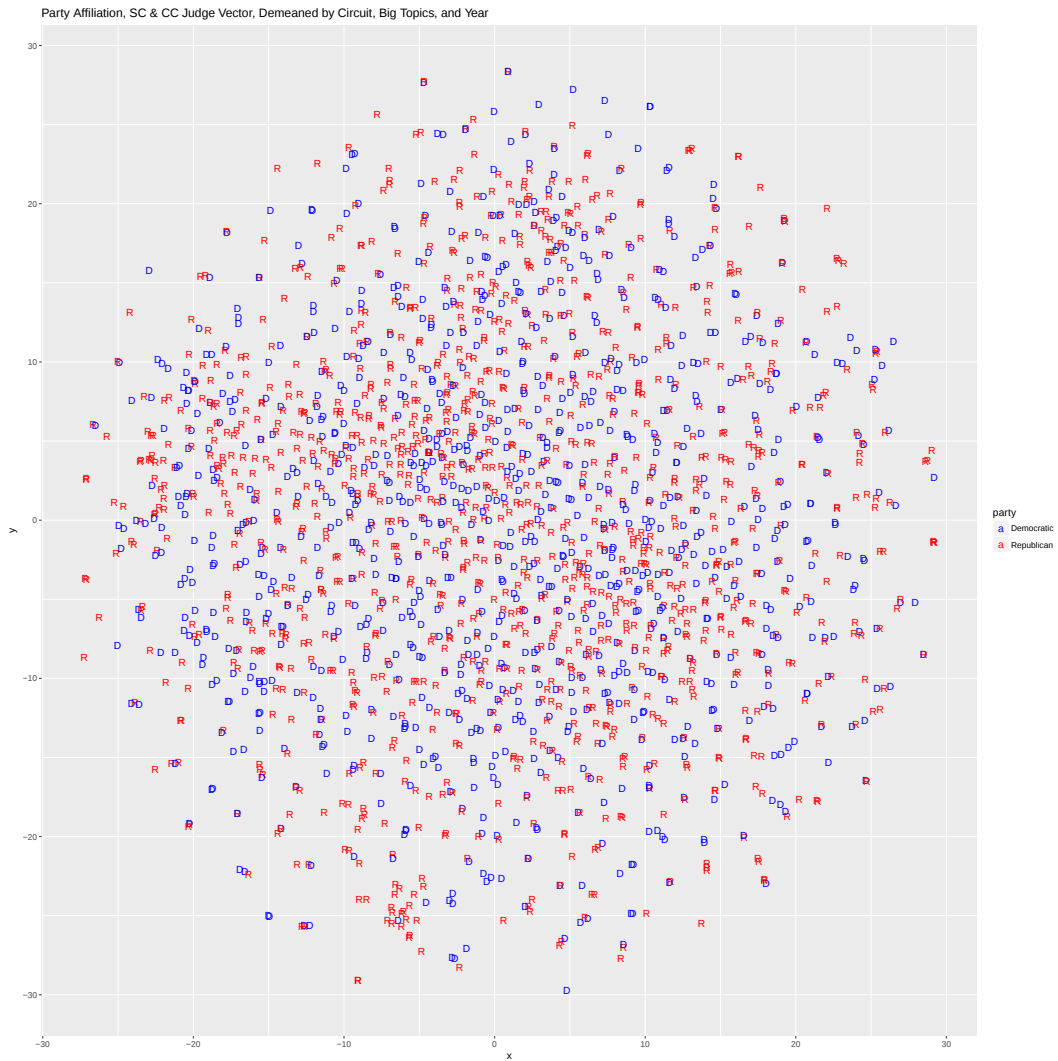


Figure 5: Centered by Court-Topic-Year, Averaged by Judge, Labeled by Judge Birth Cohort



Figure 6: Centered by Court-Topic-Year, Averaged by Judge, Labeled by Law School Attended



Table 1: Pair-Wise Similarities Between Federal Appellate Courts

	SCOTUS	1 st Circ.	2 nd Circ.	3 rd Circ.	4 th Circ.	5 th Circ.	6 th Circ.	7 th Circ.	8 th Circ.	9 th Circ.	10 th Circ.	11 th Circ.	D.C. Circ.	Fed. Circ.
SCOTUS	1.000													
1 st Circ.	0.022	1.000												
2 nd Circ.	-0.008	0.302	1.000											
3 rd Circ.	-0.001	0.135	0.207	1.000										
4 th Circ.	-0.045	-0.045	-0.081	0.126	1.000									
5 th Circ.	-0.105	-0.196	-0.298	-0.269	0.038	1.000								
6 th Circ.	-0.074	-0.185	-0.148	0.009	0.069	-0.107	1.000							
7 th Circ.	-0.097	-0.052	-0.014	-0.055	-0.162	-0.257	0.029	1.000						
8 th Circ.	-0.137	-0.215	-0.296	-0.214	-0.150	-0.184	0.050	-0.022	1.000					
9 th Circ.	0.039	-0.137	-0.140	-0.182	-0.147	-0.121	-0.220	-0.265	-0.150	1.000				
10 th Circ.	-0.111	-0.249	-0.361	-0.179	-0.189	0.017	0.006	-0.158	0.218	0.042	1.000			
11 th Circ.	-0.086	-0.191	-0.240	-0.215	0.067	0.713	-0.039	-0.224	-0.192	-0.084	0.026	1.000		
D.C. Circ.	0.846	-0.085	-0.058	0.011	-0.010	-0.062	-0.097	-0.177	-0.111	0.067	-0.025	0.011	1.000	
Fed. Circ.	0.178	0.200	0.132	0.116	0.124	-0.150	-0.154	-0.082	-0.255	-0.116	-0.260	-0.181	0.094	1.000

classification. The cosine similarity between two vectors,

$$s(\vec{v}, \vec{w}) = \frac{\vec{v} \cdot \vec{w}}{\|\vec{v}\| \|\vec{w}\|}$$

which is equal to one minus the cosine of the angle between the vectors. It takes a value between -1 and 1. In the case of word embeddings, high similarity means that the words are often used in similar language contexts.

In the case of judges, we can say that similarities approaching one mean that the judges tend to use similar language in their opinions. Similarities approaching -1 meaning the judges rarely use the same language. Similarities near zero mean that the judges are as similar to each other as would be expected from two randomly selected judges in the population.

First we look at similarity between court vectors to complement the spatial representation in Figure 1. We centered the vectors by topic and year, and then aggregated by court. We then computed the pair-wise similarities between the court vectors. These are reported in Table 1.

The colors provide a gradient for similarity, with green meaning the courts are relatively similar and red meaning they are relatively dissimilar. The table has some interesting features. First, the D.C. Circuit is most similar to the Supreme Court of the United States, which is intuitive since they are both located in Washington, D.C. and focus on issues of federal government functioning such as separation of powers. Second, the 11th circuit is similar to the 5th circuit, which is intuitive since the 11th Circuit used to be a part of the

5th Circuit and they share many legal precedents.

Next we look at similarity between judge vectors. Starting with the Supreme Court, we center the document vectors on topic, and year. Then we take the average of these centered vectors by judge as our representation of judge writing, reasoning, and beliefs. Table 2 (continued in Table 3) reports the pair-wise similarities between a selection of recently sitting Supreme Court judges. Overall, there are limited immediate insights and the results are mixed. For example, it is intuitive that Scalia is close to Thomas. But counter-intuitively, Scalia is even closer to Souter, Stevens, and O'Connor. Another example: Intuitively, Brennan is close to Thurgood Marshall; but counter-intuitively, he is closer to White and Stewart. Overall, the judge vectors do not seem to encode similarities between Supreme court judges very well. This may be due to the relatively few decisions that they author. In particular, the relative dissimilarity between Kagan and most other justices is likely due to her having only a handful of decisions in the corpus.

One interesting feature of our model is that it represents both circuit court judges and supreme court judges in the same geometric space. As done previously, we center all the document vectors on court, topic, and year. We then aggregate by judge. For Table 4, we computed the vector similarity between each circuit court judge and each supreme court judge. We then ranked the circuit court judges by this similarity. The table shows, for each supreme court judge, the top 5 circuit court judges on this ranking. As with the pair-wise similarities between supreme court judges, these rankings are not particularly intuitive or informative. Understanding the limitations of these types of models is important for future research. An important factor is that we use a bag-of-words model, and ideological differences between judges may be mostly encoded in phrases.

One reason for the lackluster results in the Supreme Court is that the judge vectors may not be well defined due to the small number of opinions they publish. Therefore we round out this analysis by looking at a notable circuit court judge, Richard A. Posner. The document vectors are de-meant by court, year, and topic. Then they are aggregated by judge. Then we rank all circuit court judges by the similarity of their vector to Posner's vector. These are reported in Table 5. Interestingly, the most similar judge is Frank Easterbrook, who, like Posner is known for the use of economic analysis in opinions. Posner has a conservative reputation, and we see other conservative judges such as Neil Gorsuch and Antonin Scalia. Henry Friendly makes an appearance – he is a well-known pragmatist, as is Posner. Finally, Michael McConnell co-write law articles with Posner. The document vectors, as trained in this example, are much more informative about the connections between circuit court judges than between supreme court judges.

Table 2: Pair-Wise Similarities between Supreme Court Judges

	AFortas	AMKennedy	AScalia	BRWhite	CThomas	DHSouter	EKagan	EWarren	FFrankfurter	FMVinson	HABlackmun	HLBlack	JGRoberts	JPStevens
AFortas	1.000													
AMKennedy	0.733	1.000												
AScalia	0.735	0.974	1.000											
BRWhite	0.834	0.908	0.913	1.000										
CThomas	0.686	0.962	0.958	0.854	1.000									
DHSouter	0.718	0.967	0.967	0.878	0.962	1.000								
EKagan	0.454	0.674	0.659	0.514	0.697	0.654	1.000							
EWarren	0.855	0.732	0.730	0.863	0.684	0.709	0.407	1.000						
FFrankfurter	0.807	0.604	0.604	0.752	0.554	0.580	0.324	0.913	1.000					
FMVinson	0.717	0.542	0.542	0.675	0.494	0.521	0.310	0.838	0.906	1.000				
HABlackmun	0.823	0.919	0.923	0.970	0.880	0.901	0.557	0.814	0.695	0.620	1.000			
HLBlack	0.873	0.706	0.706	0.847	0.655	0.669	0.381	0.943	0.930	0.854	0.803	1.000		
JGRoberts	0.569	0.869	0.861	0.728	0.862	0.850	0.679	0.575	0.445	0.389	0.734	0.521	1.000	
JPStevens	0.775	0.965	0.966	0.956	0.932	0.949	0.611	0.781	0.656	0.588	0.963	0.767	0.826	1.000
LPowell	0.818	0.908	0.910	0.980	0.852	0.882	0.508	0.841	0.720	0.643	0.975	0.819	0.725	0.958
PStewart	0.874	0.847	0.856	0.969	0.797	0.820	0.468	0.924	0.838	0.750	0.939	0.905	0.656	0.906
RBGinsburg	0.699	0.950	0.952	0.853	0.961	0.953	0.702	0.679	0.546	0.477	0.882	0.660	0.849	0.933
RHJackson	0.758	0.546	0.541	0.694	0.494	0.524	0.305	0.864	0.925	0.878	0.640	0.903	0.367	0.602
SAAlito	0.560	0.846	0.848	0.697	0.866	0.836	0.702	0.554	0.438	0.399	0.710	0.503	0.872	0.790
SDOConnor	0.761	0.964	0.962	0.950	0.930	0.944	0.588	0.777	0.660	0.591	0.955	0.757	0.802	0.976
SGBreyer	0.683	0.953	0.950	0.846	0.963	0.950	0.708	0.674	0.557	0.506	0.865	0.649	0.863	0.928
SSotomayor	0.556	0.743	0.747	0.621	0.774	0.742	0.587	0.522	0.439	0.372	0.641	0.471	0.723	0.697
TMarshall	0.827	0.900	0.898	0.962	0.857	0.876	0.543	0.832	0.717	0.626	0.968	0.830	0.725	0.948
WEBurger	0.811	0.871	0.873	0.967	0.813	0.836	0.464	0.843	0.734	0.658	0.953	0.813	0.702	0.924
WHRehnquist	0.788	0.932	0.940	0.974	0.885	0.905	0.537	0.816	0.705	0.636	0.966	0.790	0.762	0.963
WJBrennan	0.871	0.896	0.894	0.976	0.844	0.869	0.533	0.909	0.806	0.723	0.957	0.892	0.725	0.943
WODouglas	0.872	0.720	0.722	0.859	0.674	0.707	0.412	0.938	0.924	0.847	0.819	0.972	0.536	0.785

Table 3: Pair-Wise Similarities between Supreme Court Judges (cont.)

	LFPowell	PStewart	RBGinsburg	RHJackson	SAAlito	SDOConnor	SGBreyer	SSotomayor	TMarshall	WEBurger	WHRhnquist	WJBrennan	WODouglas
LFPowell	1.000												
PStewart	0.954	1.000											
RBGinsburg	0.854	0.794	1.000										
RHJackson	0.669	0.772	0.493	1.000									
SAAlito	0.688	0.642	0.860	0.369	1.000								
SDOConnor	0.946	0.898	0.921	0.597	0.778	1.000							
SGBreyer	0.841	0.786	0.959	0.492	0.885	0.926	1.000						
SSotomayor	0.621	0.598	0.745	0.365	0.741	0.708	0.744	1.000					
TMarshall	0.961	0.936	0.868	0.666	0.690	0.926	0.845	0.605	1.000				
WEBurger	0.971	0.950	0.809	0.672	0.665	0.921	0.801	0.613	0.931	1.000			
WHRhnquist	0.970	0.939	0.876	0.637	0.729	0.972	0.873	0.672	0.937	0.964	1.000		
WJBrennan	0.968	0.971	0.849	0.749	0.691	0.929	0.841	0.620	0.964	0.950	0.946	1.000	
WODouglas	0.833	0.913	0.681	0.905	0.525	0.766	0.673	0.488	0.851	0.820	0.796	0.904	1.000

4 Discussion of Future Work

We conclude with a discussion of how future work could adapt these embeddings models for empirical analysis of law.

4.1 Structured Group Embeddings

The document embeddings developed in the previous section were static, and did not explicitly model a time component. In addition, they only encoded judge identity by taking the average of a judge’s document vectors. Recent work in embeddings models seeks to include these relations more flexibly and elegantly as a part of the data generating process. Rudolph and Blei (2017) provide a model for learning dynamic embeddings, and look at how language has changed over time in the U.S. Congress over the last century. Rudolph et al. (2017) provide a model for structured group embeddings, and allow word and document vectors to have a group component and an individual component.

4.2 Vectorization of Citation Networks

The approach above used only the language of opinions to represent legal ideas. But we all know that in a common law system, the previous cases cited are a major expression of the ideological content of a decision. In future work the judge vectors could be enriched with information from the citation graph. The citations could be included as features in the document embedding. One could also treat citations as a group embedding, where a citation

Table 4: Most Similar Circuit Court Judges to each Supreme Court Judge

W E Burger	A M Kennedy	A Scalia
MARBLEY, ALGENON L. MURRAY, HERBERT F. HULL, THOMAS GRAY O'SULLIVAN, CLIFFORD DOTY, DAVID S.	SARGUS, EDMUND A., JR. NICKERSON, EUGENE H. NOTTINGHAM, EDWARD WILLIS, JR. PECK, JOHN WELD JOHNSEN, HARVEY	ROBERTS, VICTORIA A. VANCE, SARAH SAVOIA LAKE, SIMEON TIMOTHY, III SHAW, CHARLES A. O'NEILL, THOMAS N., JR.
C Thomas	D H Souter	E Warren
KEELEY, IRENE PATRICIA M. FISHER, JOE J. MCCORD, LEON SMITH, WILLIAM F. KEENAN, BARBARA MILANO	MOTZ, DIANA GRIBBON MARRERO, VICTOR DIAMOND, GUSTAVE WANGELIN, H. KENNETH BOOCHEVER, ROBERT	ZAVATT, JOSEPH C. DYER, DAVID PATTERSON SWAN, THOMAS W. WHITAKER, SAMUEL MCCORD, LEON
H A Blackmun	H L Black	J G Roberts
CORDOVA, VALDEMAR A. SINGLETON, JOHN V., JR. AGEE, G. STEVEN WHITE, JEFFREY S. DAVIS, EDWARD BERTRAND	THOMPSON, JOSEPH W. MINER, ROGER J. MACKINNON, GEORGE E. FUSTE, JOSE ANTONIO JOHNSON, ALBERT WILLIAMS	STEIN, SIDNEY H. GLEESON, JOHN WILKINS, WILLIAM W. MURRAY, HERBERT F. VAN DUSEN, FRANCIS
J P Stevens	R B Ginsburg	S A Alito
PERRY, CATHERINE DELORES GIBSON, KIM R. SNEED, JOSEPH T. JENSEN, D. LOWELL MCKEOWN, M. MARGARET	GANEY, J. CULLEN FORRESTER, J. OWEN CHASE, HARRIE B. LEAVY, EDWARD BEA, CARLOS T.	CAHILL, CLYDE S., JR. HARPER, ROY WINFIELD ELLIOTT, JAMES ROBERT HIGGINS, THOMAS A. WEST, SAMUEL H.
S D OConnor	S G Breyer	S Sotomayor
BARRY, MARYANNE TRUMP DECKER, BERNARD MARTIN WILKINS, PHILIP C. BRIGGLE, CHARLES GUY DOOLING, MAURICE TIMOTHY	SUTTLE, DORWIN W. WOODS, GEORGE E., JR. FAIRCHILD, THOMAS TEVRIZIAN, DICKRAN M., JR. WEINFELD, EDWARD	ROBRENO, EDUARDO C. PICKERING, CHARLES WILLIS SR. NUGENT, DONALD C. FARNAN, JOSEPH J., JR. LACEY, FREDERICK B.
T Marshall	W H Rehnquist	W J Brennan
VAN SICKLE, FREDERICK L. COFFRIN, ALBERT W. BOOTLE, WILLIAM A. MORTON, L. CLURE AGUILAR, ROBERT P.	MCAULIFFE, STEVEN JAMES DUNCAN, ROBERT M. KARLTON, LAWRENCE KATZ GREEN, CLIFFORD SCOTT MCNICHOLS, ROBERT J.	RESTANI, JANE A. YOUNG, GORDON E. NICHOLS, PHILIP, JR. MATSCH, RICHARD P. PUTNAM, WILLIAM LE BARON

Table 5: Most Similar Circuit Court Judges to Richard A. Posner

Circuit Judge Name	Similarity	Rank
POSNER, RICHARD A.	1.000	1
EASTERBROOK, FRANK H.	0.663	2
SUTTON, JEFFREY S.	0.620	3
NOONAN, JOHN T.	0.596	4
NELSON, DAVID A.	0.592	5
CARNES, EDWARD E.	0.567	6
FRIENDLY, HENRY	0.566	7
KOZINSKI, ALEX	0.563	8
GORSUCH, NEIL M.	0.559	9
CHAMBERS, RICHARD H.	0.546	10
FERNANDEZ, FERDINAND F.	0.503	11
EDMONDSON, JAMES L.	0.501	12
KLEINFELD, ANDREW J.	0.491	13
WILLIAMS, STEPHEN F.	0.481	14
KETHLEDGE, RAYMOND M.	0.459	15
Circuit Judge Name	Similarity	Rank
TONE, PHILIP W.	0.459	16
SIBLEY, SAMUEL	0.459	17
SCALIA, ANTONIN	0.456	18
COLLTON, STEVEN M.	0.445	19
DUNIWAY, BENJAMIN	0.438	20
GIBBONS, JOHN J.	0.422	21
BOGGS, DANNY J.	0.420	22
BREYER, STEPHEN G.	0.414	23
GOODRICH, HERBERT	0.412	24
LOKEN, JAMES B.	0.410	25
WEIS, JOSEPH F.	0.408	26
SCALIA, ANTONIN (SCOTUS)	0.406	27
BOUDIN, MICHAEL	0.403	28
RANDOLPH, A. RAYMOND	0.397	29
MCCONNELL, MICHAEL W.	0.390	30

is predicted by the other co-occurring citations, which would locate cases in a “precedent space” as well as a language space. This approach would be similar to the application in Rudolph et al. (2017), where they predicted the occurrence of a product in a grocery shopping cart based on the co-occurrence of other products. Finally, one could apply recent advances in vectorizing networks, such as node2vec (Grover and Leskovec, 2016).

4.3 Language-Based Metrics of Implicit Bias

Another future avenue in this area is the use of embeddings to extract sentiment or bias in judicial language. This work could be based on Caliskan et al. (2017), who start with an off-the-shelf word embeddings model GloVe. This pre-trained word embedding provides a representation of English-language words in a 300-dimensional vector space. They then compute similarity, which means having the same direction in the word vector space, between groups of words.

To summarize, one starts with a set of sentiment words. These could include, for example, a set of words with positive sentiment (“good”, “best”, “pleasant”, ...) and a set of words with negative sentiment (“bad”, “worst”, “unpleasant”, ...). One could take the average vector for the positive words, “pleasant” ($\vec{w}_+$), and the average vector for the negative words, “unpleasant” ($\vec{w}_-$). The idea is that the average of these vectors encodes the shared semantic component between these words for positive and negative. This shared component is likely a more accurate representation or location of these concepts in the language space.

Next, we have a set of words identifying some social distinction, such as race. The vector for “white” ($\vec{w}_W$) might include “european”, “caucasian”, etc., while the vector for “black” ($\vec{w}_B$) might include “african”, “afro-american”, etc. We then have an average vector for each social group, with the idea that the “concept” of these social groups is more accurately located in the language space. Another way to do this is to get the average vectors for names that are disproportionately given to white and black individuals (Caliskan et al., 2017). This may not work in a legal context where first names are not used very often.

Next, one can compute the cosine similarity between the two sentiment categories on the one hand, and the two social-group categories on the other. Using these metrics, one could construct a “word embedding association test” (analogous to the “implicit association test” from psychology studies) using

$$\begin{aligned}\text{Word Embedding Association Test} &= \frac{\text{White-Pleasant Association}}{\text{White-Unpleasant Association}} - \frac{\text{Black-Pleasant Association}}{\text{Black-Unpleasant Association}} \\ &= \frac{s(\vec{w}_W, \vec{w}_+)}{s(\vec{w}_W, \vec{w}_-)} - \frac{s(\vec{w}_B, \vec{w}_+)}{s(\vec{w}_B, \vec{w}_-)}\end{aligned}$$

where $s(\cdot)$ is cosine similarity. A positive value to this test means that positive-sentiment language is more closely associated to the white race, relative to the black race, in the corpus on which the word embeddings are trained. Caliskan et al. (2017) show that in a set of word embeddings trained on a broad corpus of English, there is a significant relative white-positive relation.

These types of metrics could be potentially applied to legal writings. The idea is that the text of a judge's opinions could be used to detect variation in implicit bias across judges. One could ask, for example, whether judges with a lexical bias against blacks also tend to reject discrimination complaints, or to give longer criminal sentences to blacks. One could also look for peer effects, and see whether sitting with a biased judge has an impact on a judge's decisions.

There are broader applications of lexical association available. For example, one could look at relative positive sentiment toward particular types of policies, and see whether that is associated with policy choices of the judges. One could look at gender stereotype associations, for example associating doctor with male and nurse with female. Having more traditional gender views, as detected in one's implicit language bias, might be reflected in more conservative judicial decisions related to gender rights, such as equal employment cases and gender discrimination cases.

4.4 Judge Embeddings

We saw in Section 3 that document embeddings trained from a word prediction task did not do a good job of discriminating judges on ideology. A major factor in this limitation is that the embeddings are trained just from language style of written decisions. They do not account for the direction of the decision (e.g., for or against plaintiffs). Perhaps more importantly, they do not account for the lower-court decision features. In this subsection we outline a more synthetic approach that could address these shortcoming.

To be more precise, one can move forward with the deep learning literature and directly implement an embedding layer for judge identity. Word embeddings are constructed by locating words together that are most similarly predictive for a deep learning task. In

the same way, a judge embedding could be learned by a deep learning model which locates judges together that are similarly impactful in a machine prediction task. One can use richer representations of judge characteristics besides their language, including the directions of their decisions and their citations to previous opinions. Moreover, one can let the impact of these features interact with the features of the lower-court decision being considered.

Consider the following model of judicial opinion generation. The unit of observation is an opinion i , written by judge j at time t in court/jurisdiction c . The opinion is a matrix of features Y_i , including the ruling (affirm/reverse), the text features of the opinion, and the set of citations to previous opinions. The case is a review of a district court opinion, represented by a vector of features D_i , including the text and metadata from the district court. A set of controls X_{ct} includes a range of characteristics for court and time, including some measure of the stock of precedents in court c at time t .

We would like to predict Y_i by approximating

$$Y_i \sim F(D_i, X_{ct}, j)$$

where $F(\cdot)$ is some distribution over opinion features we can approximate using a deep neural net (e.g. Goodfellow et al., 2016). Unlike the regression models that most empirical legal scholars are used to, neural nets can easily accommodate high-dimensional outcomes (such as Y_i). The model would be trained by backpropagation with stochastic gradient descent.

In particular – and this is the key innovation – the judge identity j will be represented with an embedding lookup layer to a relatively low-dimensional dense vector space. The location of the judge vectors, initialized randomly, would be endogenous to the model. As the model goes through further training, the locations of these vectors will be pushed around to improve predictiveness. As a by-product of the model, the judges that locate together in the vector space would be predicted to behave similarly on the court holding other factors equal.

This model could then be used to simulate counterfactuals. For example, how would the decision in a case change by switching out the authoring judge j ? How would the style of language change for a different circuit c ? This will give us new insight into the topography of ideology in the U.S. judiciary.

References

- Ash, E., Chen, D., and Liu, W. (2017). The (non-)polarization of u.s. circuit court judges, 1930-2013. Technical report. 3.4
- Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4):77–84. 1, 2
- Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., and Kalai, A. T. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357. 2
- Caliskan, A., Bryson, J. J., and Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186. 4.3
- Cameron, C. and Kornhauser, L. (2017). What courts do . . . and how to model it. Technical report, NYU Law and Economics Research Paper. 1
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>. 4.4
- Grover, A. and Leskovec, J. (2016). Node2vec: Scalable feature learning for networks. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 855–864, New York, NY, USA. ACM. 4.2
- Jurafsky, D. and Martin, J. H. (2014). *Speech and language processing*, volume 3. Pearson London. 1
- Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*. 1, 2, 3.2
- Levy, O., Goldberg, Y., and Dagan, I. (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics*, 3:211–225. 1, 2
- Maaten, L. v. d. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605. 3.4

- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119. 1
- Rudolph, M. and Blei, D. (2017). Dynamic bernoulli embeddings for language evolution. *arXiv preprint arXiv:1703.08052*. 4.1
- Rudolph, M., Ruiz, F., Athey, S., and Blei, D. (2017). Structured embedding models for grouped data. In *Advances in Neural Information Processing Systems*, pages 250–260. 2, 4.1, 4.2