Supreme Court Verdict Prediction

Team Members: Maggie Chen, Carolyn Liu, Eujene Yum

Goal of Project

Our group is interested in using the US Supreme Court oral arguments to predict case outcomes. This seems like a relevant topic with important decisions being made today such as the reversal of Roe v.s. Wade and rulings on abortion pill availability per state. This topic seems like a good way to use NLP since our model will consume massive amounts of documents that humans cannot efficiently comprehend. We will use the data from Convokit (Cornell Conversational Analysis Toolkit) which contains approximately 1,700,000 utterances over 8,000 oral arguments transcripts from 7,700 cases. We will be using Pytorch to conduct our analysis.

Literature Review

Dickinson (2019) retrieved oral argument transcripts of nearly 1,400 cases argued and decided from 1998 to 2015 from the Supreme Court's website. The transcripts were paired with case outcomes from SCDB (Supreme Court Database) as labels. The research adopted four categories of features: question count features, question chronology features, question sentiment features and N-Gram features. Stanford CoreNLP sentiment annotator was used to get the "question sentiment features". An exploratory analysis of the features show that almost all the Justices ask longer questions to the party they finally vote against, the tone is less friendly and more follow-up questions would be involved. EDA proves the effectiveness of the selected features.

Given each Justice has a different questioning style, the research created separate feature matrices for each Justice and the SVM model (Support Vector Machine) was built upon feature matrices. Since every N-Gram is represented as a column, the feature matrix is extremely wide and SVM was chosen because of its effectiveness in dealing with high dimensional feature spaces.

A ten-fold cross validation was used throughout the model training process. Accuracy was adopted as the evaluation metric without much explanation of the reasons behind. The SVD model, with all four categories of features, correctly predicted individual Justice votes in 73% of the cases, with the highest accuracy of around 86.8% for Justice Rehnquist and the lowest of around 52.3% for Justice Sotomayor. Dickinson also examined the performance of each category of features. Though the category of "N-Gram features" accounted for the vast majority of features, each category of features generally added some incremental predictive accuracy beyond the N-Grams alone.

This research paper includes features such as “question count features” which are not typical in other papers, thus providing us with alternative features that can be adopted in our project. The research also emphasized different questioning styles of each Justice and built one model for each Justice. This inspired us on if we should do a case outcome level prediction, a justice vote level prediction, or compare them side by side. One caveat of this research is that the feature matrix may turn out to have more columns than rows, given each N-Gram is represented as a column. Since p>>n violates the common assumption in Machine Learning and might bring overfitting issues, we would try to avoid it in our own project.

Aletras et al. (2016) uses NLP techniques to predict whether specific articles of the European Convention on Human Rights have been violated as decided by the European Court of Human Rights, given text extracted from the case. The Court judgments have a distinctive structure: procedure, the facts (case circumstance, relevant laws), etc.

The authors obtained data from HUDOC, parse for English cases, then achieve a balance between classes. Text was extracted using regular expressions and exclude any information that directly pertain to the outcome of the case.

For each section of the case, the authors derived n-grams features and topics from the text extracted from each section. For each set of cases, the top-2000 most frequent N-grams where $N \in \{1,2,3,4\}$ were computed. Each feature represents the normalized frequency of a particular N-gram in a case or a section of a case. They then “extracted N-gram features for the Procedure (Procedure), Circumstances (Circumstances), Facts (Facts), Relevant Law (Relevant Law), Law (Law) and the Full case (Full) respectively (note that the representations of the Facts is obtained by taking the mean vector of Circumstances and Relevant Law)”. Topics for each article were then created by clustering together N-grams that are semantically similar by leveraging the distributional hypothesis suggesting that similar words appear in similar contexts.

The authors used SVM with a linear kernel in order to identify important features that are indicative of each class by looking at the weight learned for each feature to predict the Court’s decisions. Additionally, they tuned the regularization parameter C using grid search. To evaluate model performance, the authors computed the mean accuracy obtained by 10-fold cross-validation. The author’s achieved < 70% classification accuracy for most different feature types across articles, with ‘Circumstances’ n-grams giving the highest accuracy for Articles 6 & 8. We can potentially adapt the authors’ use of n-gram features for each section of the case to our project.

DM (2017) constructs a model designed to predict the behavior of the Supreme Court of the United States in a generalized, out-of-sample context. They develop a time-evolving random forest classifier that leverages unique feature engineering to predict more than 240,000 justice votes and 28,000 cases outcomes over nearly two centuries (1816-2015). They achieve 70.2% accuracy at the case outcome level and 71.9% at the justice vote level. Over the past century,

they outperform an in-sample optimized null model by nearly 5%. Their model is distinctive because it can be applied out-of-sample to the entire past and future of the Court, not a single term. Our results represent an important advance for the science of quantitative legal prediction and portend a range of other potential applications.

This paper uses data from the Supreme Court Database, and each case contains as many as 240 variables, many of which are categorical. The target variable can have 3 values – reverse, affirm, or other. The paper only includes cases that reverse or affirm. They also convert categorical variables into multiple binary variables.

DM (2017) focuses on building a general model—one that could stand the test of time across many justices and many distinct social, political and economic periods. They construct their model using a random forest classifier, which does not generally require pre-processing. They mention that one drawback of the scikit-learn implementation of random forests relative to alternatives like xgboost, is that it handles missing data by mapping missing values to a separate "missing" indicator column during encoding, but a historical mean imputation is used sometimes. The paper states that random forests are very effective and have proven to outperform support vector machines and multi-layer perceptron models. DM (2017) experiments with "growing" forests where they add onto the existing forest with new terms and a "fresh" forest model where new forests are created each term with a number of trees selected by cross-validated hyperparameter search. The "growing" approach allowed for faster simulation time and stable prediction, and cross-validation and hyperparameter search did not have a noticeable impact on accuracy over "default" random forest configurations.

It will be helpful for us to take this information into consideration as we build a random forest model to conduct our own predictions on Supreme Court outcomes.

Plan of Action

By mid-quarter, we would like to have preprocessed our data, done feature engineering, conducted exploratory data analysis as well as run our baseline models which would include logistic regression, SVM, and a tree-based model. We will evaluate our models' accuracy by comparing to the actual outcome of the court case. We will also compare our models against those in our literature review. We will most likely work together on most models, with potentially each person taking ownership of tuning specific models.

References

Aletras, Nikolaos, Dimitrios Tsarapatsanis, Daniel Preoţiuc-Pietro, and Vasileios Lampos. "Predicting judicial decisions of the European Court of Human Rights: A natural language processing perspective." *PeerJ Computer Science* 2 (2016): e93.

Dickinson, Gregory M., A Computational Analysis of Oral Argument in the Supreme Court (July 3, 2019). 28 Cornell J.L. & Pub. Pol'y 449 (2019), Available at SSRN: https://ssrn.com/abstract=3198401

Katz DM, Bommarito MJ 2nd, Blackman J. A general approach for predicting the behavior of the Supreme Court of the United States. PLoS One. 2017 Apr 12;12(4):e0174698. doi: 10.1371/journal.pone.0174698. PMID: 28403140; PMCID: PMC5389610.