Confidential Informants Project

CS506 Spring 2020

Outline Of Report

I. Project Overview

The goal of the project was to analyze all criminal cases within the state of Massachusetts in order to find instances and patterns of when the state had used informants improperly in criminal cases.

Informants are witnesses who may at times testify anonymously. For the purposes of our project, the term "informant" is synonymous with "confidential informant". Confidential informants are sometimes criminals who are freed from jail or offered lighter sentences in exchange for their collaboration with law enforcement. The state of Massachusetts usually argues that confidential informants must have their identity protected in order to keep them safe from reprisals, but this obviously leads to potential for abuse. A motivation for this project was several high profile cases uncovered by WGBH and other news outlets in which the state had abused the confidential informant system to bribe inmates into lying to obtain false convictions in cases where there was little other evidence.

This project was requested by Paul Singer, head of investigative reporting for WGBH.

II. Data Collection

A. Data We Started With

Per direction from the client, we began our analysis with data from project teams that worked on a similar project in 2018 and 2019. This data, contained in the .json files cases.json and mass_appeals.json, represented the decisions of all *published* cases from the Massachusetts Supreme Judicial Court (SJC) and Massachusetts State Appeals Court respectively in the 2008-2018 period.

For an example of the type of dense legal language used in the decisions, see the following, which is the beginning of a case:

"'The present case is the most recent in a series of cases concerning the egregious misconduct of Annie Dookhan, a chemist who was employed in the forensic drug laboratory of the William A. Hinton State Laboratory Institute (Hinton drug lab) from 2003 until 2012. On January 23, 2007, the defendant, Admilson Resende, pleaded guilty on indictments charging distribution of a class B controlled substance (cocaine), G. L. c. 94C, § 32A (c) (five counts); violation of the controlled substances laws in proximity to a school or park, G. L. c. 94C, § 32J (three counts); and possession of a class B controlled substance (cocaine) with intent to distribute, G. L. c. 94C, § 32A (c) (one count).He completedPage 3service of his sentences.On October 2, 2012, the defendant filed in the Superior Court a motion to withdraw his guilty pleas pursuant to Mass. R. Crim. P. 30, as appearing in 435 Mass. 1501 (2001), based on Dookhan\'s malfeasance.Prior to the issuance of a ruling on the defendant\'s motion, this court decided Commonwealth v. Scott,(2014), in which we articulated, in reliance on Ferrara v. United States, 456 F.3d 278, 290-297 (1st Cir. 2006), a two-prong framework for analyzing a defendant\'s motion to withdraw a guilty plea under rule 30 (b) in a case involving the misconduct of Dookhan at the Hinton drug lab. Scott, supra at 346-358. Under the first prong of the analysis, a defendant must show egregious misconduct by the government that preceded the entry of the defendant\'s guilty plea and that occurred in the defendant\'s case."

B. How we Scraped

Within the initial data, it became clear that there were only about 96 cases that contained informants (see the analysis section). Thus, on the direction of Mr. Singer, we decided to scrape about 10 more years of cases, from 2000-2008. We ran into problems when we realized that the site that all three previous teams had used to scrape court decisions was no longer operational. We eventually found the somewhat equivalent site *masscases.com*--data there is not labelled as thoroughly as data from the site that was previously used, and in addition to having to completely rewrite the code used for scraping because of the different site, the lack of tagging made scraping more difficult. We decided to use selenium webdriver to accomplish the scraping, and once the code was written, it took only a few hours to execute.

III. Data Cleaning

A. How we Combined the Data

The new data was scraped to have a similar format to the old data, so merging the two data sets would be easier. We settled on pandas DataFrame for the final structure, as the efficiency and implementation of it compared to python lists is vastly superior. With this change we had to reformat the inner fields of the dictionaries to strings instead of lists. This makes it easier for pandas to search for and work with the data. This change led to a necessity for new cleaning functions and new search functions that implemented pandas and worked with our new data structure. Now, a search for cases that contain informants, or cases that satisfy other criteria, is easy returns the full cases in a new pandas DataFrame, which is in turn searchable and has all the usability of pandas. This is a significant improvement in terms of searching the intersections and finding more complex feature groups in the data.

B. How we Cleaned the Data

1. Dissents / footnotes

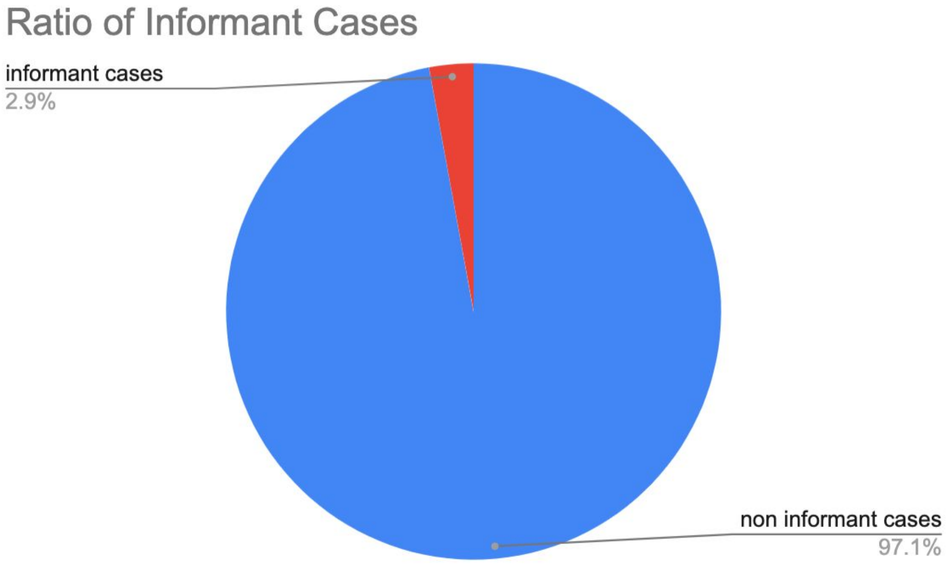
We noticed that the decision of many cases also includes a dissent, or contradictory, implemented opinion of the minority of judges, and the dissent or a footnote in our cases can alter the results of our analysis. This is due to the fact that they may contain keywords that are not relevant to the outcome of the case. While scraping the data we were careful with the footnotes and tried not to include any, regrettably, we think we still have some left. This is due to the fact that we searched for the keyword “Footnote” and found 500 examples of it in our data. However, distinguishing these footnotes from citations is difficult.

2. The Issue of Punctuation

We found that the data was formatted in such a way that there were no spaces between words and punctuation so when the data was put into word2Vec the punctuation would throw off the algorithm. For example, “informant” and “infromant.The” were considered two different words. We fixed this issue by editing all of the text to have spaces on each side of any punctuation or special characters.

IV. Data Analysis

A. How we located cases that contain informants



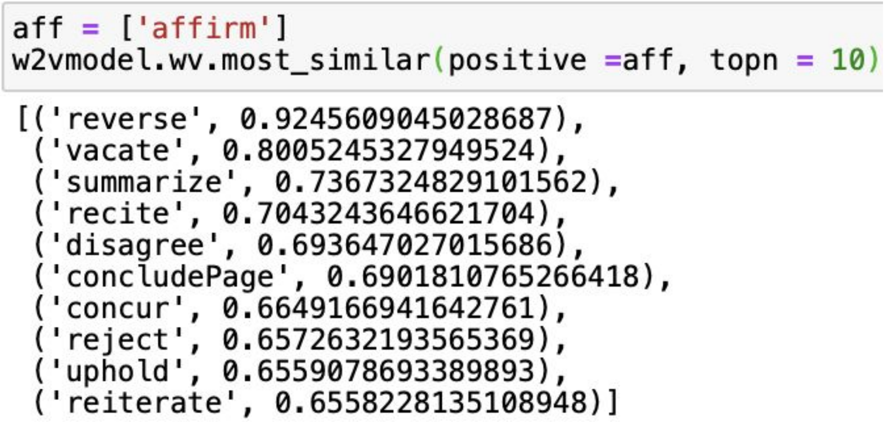
Initially we were looking at cases that had informants by simply searching through the data for the keyword “informant”, we realized we needed to expand our search and found a list of keywords that were related to informants., including "confidential informant", " ci ", "snitch", "informant", and "gang informant".

This expanded our search, but the amount was still low. Our search returned 79 cases in our SJC cases, and 146 cases in our appeals court cases. The graph shows how these cases make up only 3% of our dataset.

B. The state of our overturned analysis

To answer the key question of which cases--informant cases and otherwise--were overturned, we started by searching through cases to understand the language that they used in order to determine if the case is to be overturned or not. This preliminary analysis showed that the verdict was not in the header, but in the text. The principal verbs that signified a verdict were "reverse" and "affirm", however these were almost always ambiguous, as they would sometimes each appear multiple times in each case, often in the form of citations of previous cases with that status. After reading several cases, I formed a sort of "manual decision tree" to determine case status--keywords "We reverse" and "We affirm" were least ambiguous, and then cases with "affirm" and "reverse" used unambiguously could be analyzed, and then a few other keywords.

Keywords to use in this analysis were informed by a word2vec analysis of what keywords were used similarly in the cases to "reverse" and "affirm"--see below for one example of the output of this technique (note that similarity here is a sort of "contextual similarity"--"affirm" is most similar to "reverse" because they are both used to make conclusions about cases).

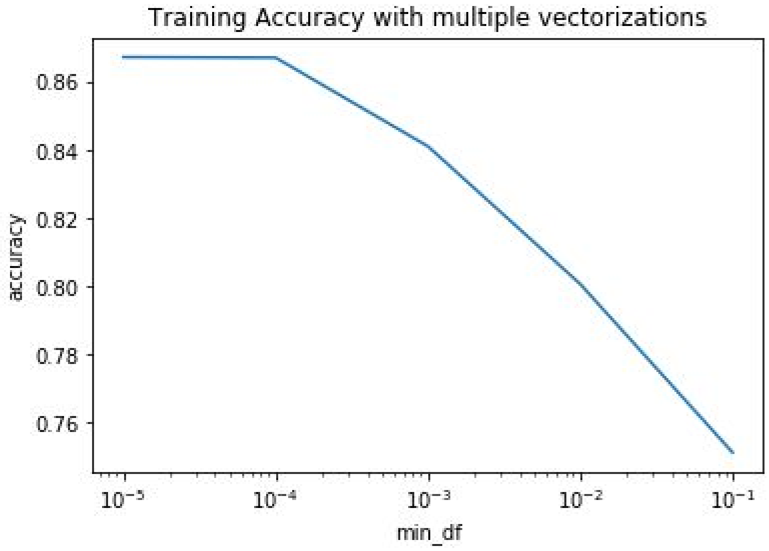


```
aff = ['affirm']
w2vmodel.wv.most_similar(positive =aff, topn = 10)
```

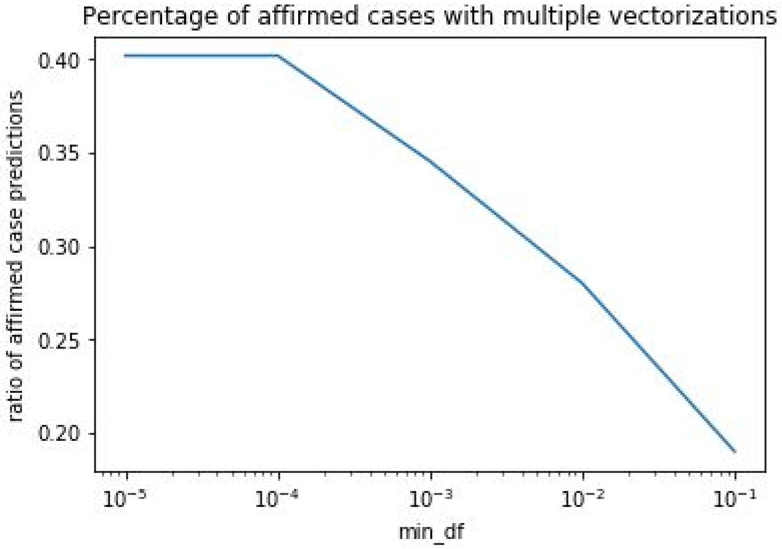
```
[('reverse', 0.9245609045028687),
 ('vacate', 0.8005245327949524),
 ('summarize', 0.7367324829101562),
 ('recite', 0.7043243646621704),
 ('disagree', 0.693647027015686),
 ('concludePage', 0.6901810765266418),
 ('concur', 0.6649166941642761),
 ('reject', 0.6572632193565369),
 ('uphold', 0.6559078693389893),
 ('reiterate', 0.6558228135108948)]
```

C. *The state of our overturned predictions and our logistic regression model :*

With our "manual decision tree" improved by our word2vec analysis we were able to classify 80% of the cases as affirmed or reversed. For the last 20% we employed a logistic regression model to predict if the cases were overturned or not. We trained the model on a vectorized form of our text data from the data we have labels for using our analysis up to this point of which cases are overturned. Our vectorization came from the TFIDF vectorizer with a minimum word frequency of 10^{-3}. This hyper-parameter was tuned by training the model on 5 different orders of magnitude of this parameter, ranging from 10^{-1} to 10^{-5}, the last two orders encompassing the whole dataset. We avoided taking the last two values as we presumed the model would start overfitting at this point.



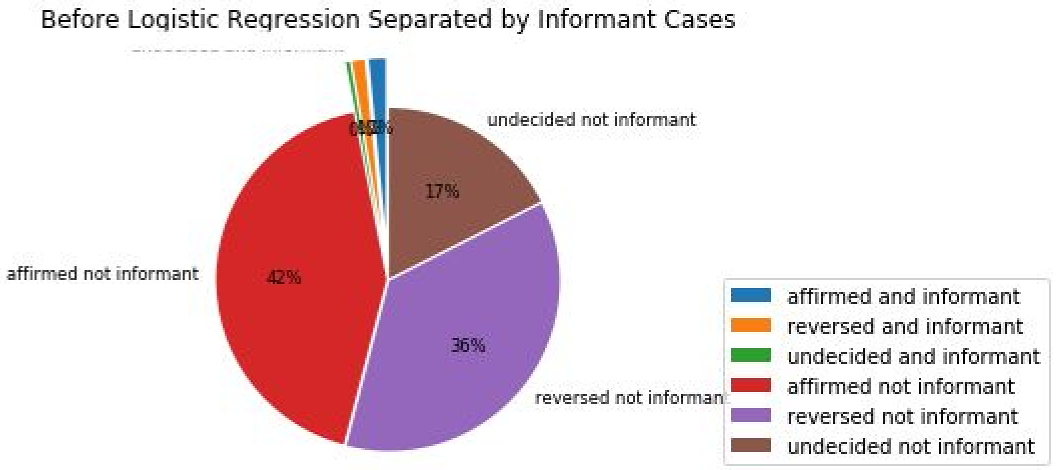
It is interesting to see that the percentage of affirmed cases also drops in a similar manner as the parameter is increased, maybe some of the relevant information of affirmed cases is in fact in some obscure terms that are not frequent, but this is not probable, and therefore a second reason not to use the two smallest hyperparameter values.

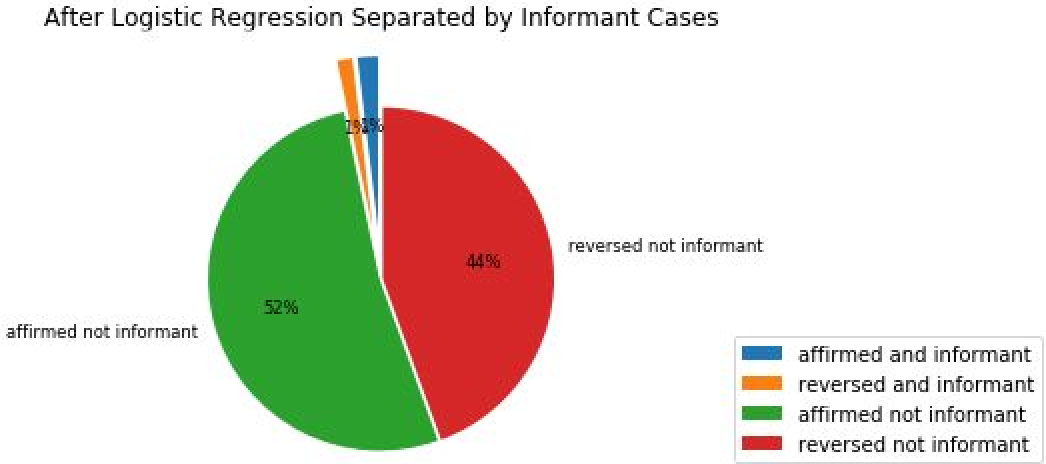


Once the parameter was fixed, we achieved an accuracy of 86% on our training data, which we deemed high enough to interpolate labels for the data we were otherwise unable to classify. It returned 503 affirmed cases and 751 reversed cases, this distribution is within a small margin of the distribution of the original data without logistic interpolation of the data.

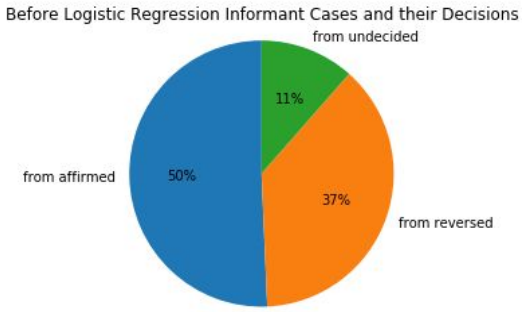
D. *How much more often are informant cases overturned?*

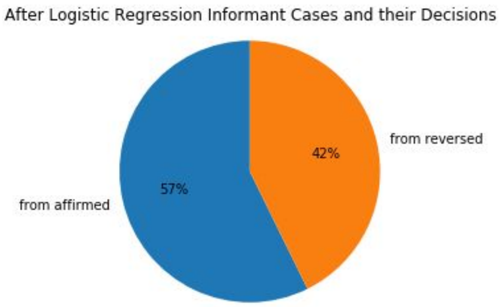
During our analysis we focused on the verdict of cases in order to give us insights on how informant cases play out in court. We focused specifically on the keywords that we found in word2vec and the words we had found useful in our decision tree, as we found that the cases mentioned these words to produce their verdict. It is interesting to see that our logistic regression model used to interpolate for cases we could not otherwise analyze did not shift the results of whether informant cases were overturned more, rather it completed the results by interpolating for the unknown cases. These next graphs show the total dataset proportions of cases divided in affirmed, undecided, and reversed, but also by cases that contain informants or not. These are before and after we used the logistic regression model to predict the undecided cases.

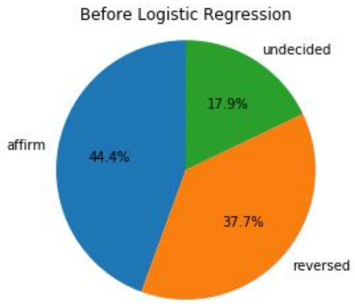

Before Logistic Regression Separated by Informant Cases

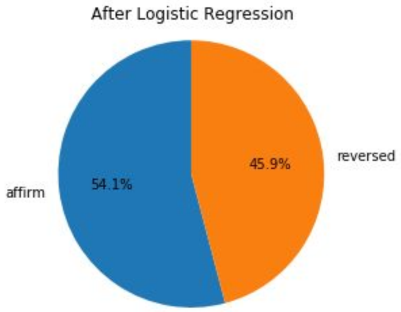

After Logistic Regression Separated by Informant Cases

These next graphs show the proportion of reversed and affimed cases in the informant subset of cases, as well as in cases as a whole, in order to compare how often cases are overturned in these categories


Before Logistic Regression Informant Cases and their Decisions


After Logistic Regression Informant Cases and their Decisions


Before Logistic Regression


After Logistic Regression

The first row of pie charts shows informant cases with and without logistic interpolation, and the second row, all cases, both SJC and State Appeals Courts cases, with and without interpolation. It is to our great disappointment that we show that there is no correlation that shows that having informants will result in a higher likelihood of a case being overturned.

Nevertheless, we have narrowed the search from 7000 cases, to 120 that are overturned and have informants, to search for newsworthy material for Mr. Singer, as we will discuss later on.

E. *What are the factors in cases being affirmed/reversed?*

Since we made a logistic regression model of when cases are affirmed versus reversed, we can comment on what words and concepts generally correspond to being reversed and affirmed--we were hoping informants would be part of this, but they were not.

We make this analysis based on the coefficients that define the decision hyperplane of our logistic regression model: here, a large negative coefficient means that a word correlates strongly to a case being reversed and a large positive coefficient correlates to a case being affirmed (note we list only the 30 coefficients of largest absolute value, as these are the most important). These coefficients are directly comparable, as the frequency outputted by the TFIDF Vectorizer is normalized but not mean centered. Therefore, the input space is homomorphic where every dimension is in the same unit.

reversed -2.491500	affirm 6.028632
reverse -2.369554	211 2.208358
remanded -1.871203	petition 1.813208
complainant -1.644267	claims 1.529959
ordered -1.376633	justice 1.456505
proceedings -1.359415	relief 1.442526
question -1.252139	discretion 1.403535
officer -1.190199	argues 1.361432
case -1.157154	murder 1.349380
remand -1.145449	claim 1.326886
note -1.139534	33e 1.312791
williams -1.124071	abuse 1.303382
appellate -1.085217	contributions 1.278954
presentment -1.071989	single 1.251117
self -1.039116	title 1.241139

While it is likely surprising to have numbers as part of this list, “211” was usually used as part of the citation “G.L.c. 211”, which refers to a Massachusetts General Law chapter 211, entitled "THE SUPREME JUDICIAL COURT", which describes the legal bases for the Supreme Judicial Court and also the conditions for emergency appeals. Appeals were often argued for under this law, and therefore judges would bring it up when denying them. Likewise, for “33e”, judges would bring up “G.L.c. 278 s. 33e”, a short subsection of Massachusetts trial court law[1]

[1] The full text of this section reads as follows:

"Section 33E. In a capital case as hereinafter defined the entry in the supreme judicial court shall transfer to that court the whole case for its consideration of the law and the evidence. Upon such dealing with how capital cases should be tried, to affirm decisions in capital cases.

F. *Important Result: Notable Cases*

In the GitHub repository in the Data folder there is a file of cases titled “Interesting_cases.CSV.” In the file, there is a list of cases with all the information we have on said cases. We recommend to our clients look through these cases as we have identified them as being the most likely ones to produce a story. The bases of this claim comes from each of the cases having informats being the main weight on the decision, and there to be some room for doubt on the validity of the informant.

G. *Summary of the Role of word2vec in our Project*

After finishing the scraping of the new data, we looked at different ways we could vectorize the data. After doing research on other projects similar to our own online, emailing with Gowtham, and speaking with professor Galletti, we chose that Word2Vec was the best way to vectorize our data. We took in the 4 JSONs of cases and created a tokenization of all the text from all the cases, keeping them separated between SJC and Statewide Appeals Court cases. We ran into a slight issue regarding punctuation once we put this data into Word2Vec. The fashion in which we fixed this is described above. We fixed this issue by editing all of the text to have spaces on each side of any punctuation or special characters. Once this cleaning was finished, we found that word2Vec provided a great system to discover more about our data. We have discovered multiple words that are used in a similar fashion as “affirm’ and “reverse” to now use in future scraping schemes. Going forward, we are going to use this word2Vec embedding to do more advanced analysis on our data.

consideration the court may, if satisfied that the verdict was against the law or the weight of the evidence, or because of newly discovered evidence, or for any other reason that justice may require (a) order a new trial or (b) direct the entry of a verdict of a lesser degree of guilt, and remand the case to the superior court for the imposition of sentence. For the purpose of such review a capital case shall mean: (i) a case in which the defendant was tried on an indictment for murder in the first degree and was convicted of murder in the first degree; or (ii) the third conviction of a habitual offender under subsection (b) of section 25 of chapter 279. After the entry of the appeal in a capital case and until the filing of the rescript by the supreme judicial court motions for a new trial shall be presented to that court and shall be dealt with by the full court, which may itself hear and determine such motions or remit the same to the trial judge for hearing and determination. If any motion is filed in the superior court after rescript, no appeal shall lie from the decision of that court upon such motion unless the appeal is allowed by a single justice of the supreme judicial court on the ground that it presents a new and substantial question which ought to be determined by the full court." (See https://malegislature.gov/Laws/GeneralLaws/PartIV/TitleII/Chapter278/Section33e)

V. Caveats About the Interpretability of the Data

A. Note on Massachusetts court system

The Massachusetts Court system is made up of several levels, with the SJC as the highest court of final appeal and the Statewide Appeals Courts as the court system immediately below that in importance and in line of appeal. While in some cases the SJC may exercise what is known as its "right of superintendency" and chose to hear cases from much lower courts and skip over the Statewide Appeals Courts, most of the cases in the SJC are further appeals from the Statewide Appeals Courts and are thus duplicates in some sense. We have set aside the problem of figuring out which cases are duplicates in this sense because there is not sufficient metadata in the pages scraped to do this unambiguously without a complex algorithm that would involve cross referencing data and the names of the parties with an additional website containing records of what cases were tried on which date at which courthouse.

For this reason, we often analyzed cases from the SJC and from the Statewide Appeals Courts separately, but not always, as at times this would have reduced the size of our dataset too much.

B. Caveats About Unpublished Cases

Note that as well, we only had access to *published* cases. Some fraction of cases are selected as not being important enough, or not modifying precedent enough, to have their decisions recorded in official legal books or to have their decisions published online. It is unknown what the ratio of published to unpublished cases are, though from anecdotal evidence in talking to the law student Betsy Byra and searching specialized expensive databases which contain unpublished cases (but which cannot be scraped without breaking Terms of Service), unpublished cases seem somewhere between a quarter and a third. A source with significant experience in the Massachusetts legal system was unable to give us more information on this point.

VI. Challenges still to be addressed

A. More advanced overturned analysis

For a more accurate result, it would be good to minimize the amount of cases that need to be predicted by logistic regression. It would be a good idea to do this mainly with domain-specific research, as this was a weakness of our project. As well, it would be possible to important words that define the decision hyperplane[2] on the logistic regression, and the most similar words given by word2vec. One could then look at the intersection of these, as those have a more reliable relationship to the keywords that define the verdict. After the fact, it is necessary to have a proof of concept by reading the last cases whose verdicts were predicted, and those of which only contain one keyword, especially if this keyword is new, to again make sure that these highlighted keywords make sense in human language.

B. *More advanced word2vec*

Going forward, we would like to use our word2vec scheme to be used as an embedding system for more complex analysis. We will be using word2Vec to compare known suspicious cases to other informat cases to see if we can make predictions about the validity of their ruling.

C. *Logistic Regression*

Our logistic regression analysis of coefficients would be more sophisticated if we had had time to include bigrams as features, this would triple the initial size of the feature space, but it could be reduced to better and more meaningful components as well.

D. *Find Sophisticated Duplicates*

Find duplicates between SJC and State Appeals Cases--a contact within the legal system noted this can be done most effectively with the so called "probation number" of the person on trial.

[2] The boundary, so to speak, between keyword frequencies at which the logistic regression makes a decision.