Kent School of Law Project

Thomas BOOT[1]

[1]IIT, Illinois Institute of Technology, 3300 S Federal St, Chicago, IL 60616

tboot@hawk.iit.edu

Abstract – The Kent School of Law project has for main goal to help professionals in the law department with a cognitive and reliable data science approach to facilitate the research before a case. Indeed, the idea is to find patterns among a huge data set in order to find potential trial outcomes based on previous ones that occurred under specific judges who expressed a verdict regarding the profile of the case. Many possibilities are worth considering. In this paper, we will establish the ground work done in order to achieve this long-term goal. From data preprocessing, passing by data analysis and information extraction, this project englobes many powerful and reliable data science techniques in order to perform graph analysis, sentiment analysis, cluster analysis and outcome prediction of a trial.

1. Problem Statement

Nowadays, within a context where data science is considered to bring valuable information to companies, Machine Learning and Deep Learning revealed to be of a great usefulness in many fields. For instance, in the law department, one question could pop up being, how could we predict the outcome of a trial based on the deep analysis of previous cases similar to one of the lawyers' client. In an ideal case, they could learn from law data, valuable information regarding the similarity of the case in question with sealed ones, the prediction of the success percentage of winning the trial, and many more. The number of possibilities are without limit and it could bring a lot of benefits for the lawyers.

Law publication and opinions are available over the internet, and our input will be to extract valuable information from those files, and structure the data thanks to several analyses.

2. Free Law Project

2.1 Introduction

The Free Law Project, founded in 2013, supports open access to legal documents and to develop open-source legal research tools.

It has a number of initiatives, including:

- CourtListener, that is a growing repository of court decisions reaching over 3 million legal opinions from 419 jurisdictions, including real time updates from all US appellate courts, an archive for the appellate courts back to the 1940's and a complete US Supreme Court records since 1754.
- A judge and appointer database, which provides biographical and electoral information about 8,000 American judges and appointers.
- Over 800,000 minutes of oral arguments.

We will start our work from scratch, and access our data thanks to the API-accessible website of CourtListener.

2.2 Goal of our project

As a recall, the big picture of this project is to predict the outcome of a trial based on patterns from judges on previous case from the past. Indeed, for this research paper, we are focusing on specific jurisdictions, to begin with SCOTUS (Supreme Court of The United States). Our goal is to apply opinion graph analysis, sentiment analysis, cluster analysis, in order to predict the outcome of a case's verdict.

3. Data Overview

3.1 Data Source

Every jurisdiction from the database is organized under JSON files and will be considered as our raw data. All opinions from all jurisdictions have a size of 36 GB. Unable to process all this data on one computer, the idea was to run all our experiments with AWS, when we first have our idea of our approach for this project. During this project, our data will be a sample of 5,000 opinions from the SCOTUS jurisdiction. A header of one JSON file would contain the following features:

absolute_url, resource_url, author,autor_str, cluster, date_created, date_modified, download_url, extracted_by_cor, html, html_lawbox, html_with_citations, joined_by, local_path, opinions_cited, per_curiam, plain_text.

Even before analyzing the content of those features, it is already clear that we would have to parse our data before processing. The packages used for each step of our project will be explained further.

3.2 Architecture of the project

The parsing and cleaning of our raw data will lead to the following approach. First, we proceed with specific feature extraction. Indeed, during our project, it was recurrent to see JSON file features incomplete, therefore we narrowed it down to 3 features: *plain text*, *HTML and HTML with citations*, where we could find our opinions about a certain case. Once cleaned and structured we will performed citation clustering based on the visualization of our network of opinions/citations. The visualization would reveal important information about how all those opinions are classified and if they would correspond to a specific cluster that we could potentially label. Besides, some additional experiments will be performed to provide the outcome of a case thanks to text mining and observation labeling.

3.3 Programming Language, packages

This project will be divided into two programming languages, Python and R.
The extraction of the data from the JSON files is done in Python. For reading those JSON files, we used json to parse the data.

To scrap the *HTML/HTML with citations* features, we used a package BeautifulSoup that allows to navigate into a html file really easily, and locate specific tags in order to extract the corresponding text.

For graph visualization and processing, we will use the following library : networkx

Those libraries will be helpful for Laplacian matrix generation and spectral clustering analysis.

For the text processing, we will use NLP packages in order to work on data preprocessing, clustering and classification. Those libraries in question are tidytext and sentiword. Sentiword is a lexical resource for opinion mining and tidytext is an effective way to clean data and make it easy to manipulate, model and visualize the data.

The Stanford OpenNLP libraries have distinct features including tokenization and part-of-speech tagging, this will be used in the name entity recognition among the other scraping methods.

4. Experiments

4.1 Feature extraction

The main task was to deal with the inconsistency of the data, as already mentioned above, our data isn't cleaned and is unstructured, therefore we take into consideration these three features: *plain text*, *HTML* and *HTML with citations*.

The structure of the html feature has to be exported to a data frame in order to proceed with the processing of the data. Figure 1. Shows the structure of the html feature for one specific opinion.
All opinions having a missing html field were ignored for the analysis.

```
'<p class="case_cite">544 U.S. 917</p>\n    <p c
lass="parties">BERWICK<br><i>v.</i><br>UNITED ST
ATES.</p>\n    <p class="docket">No. 04-8529.</p
>\n    <p class="court">Supreme Court of United
States.</p>\n    <p class="date">March 21, 2005.
</p>\n    <div class="num" id="p1">\n      <span
class="num">1</span>\n      <p class="indent">C.
A. 2d Cir. Reported below: 107 Fed. Appx. 253. M
otions of petitioners for leave to proceed <i>in
forma pauperis</i> granted. Certiorari granted,
judgments vacated, and cases remanded for furthe
r consideration in light of <i>United States</i>
v. <i>Booker,</i> 543 U. S. 220 (2005).</p>\n
</div>\n    '
```

Fig 1. Html feature of an opinion for SCOTUS. The highlight sentence shows the structure of a case cited.

For the plain text feature, we used a similar approach as for the html feature, using the package BeautifulSoup, but here the difference lies within the specific tags.

4.2 Citation Clustering

Once the specific data has been cleaned and scraped into specific data frames, we were able to run our first experiment of citation clustering.

A case cited is stated as for example:

410 US 113 (1973)

The citation is read:

- 410 : Volume 410 of the
- US : United States Report

- 113 : case begins on page 113
- 1973 : date of the case decision

The idea for the clustering experiment is to cluster all the opinions based on the citation they cite.
As opinion may call several cases, we have to scrap and extract the useful information for clustering. For every document (opinion), we select its ID and associated with it the specific volume number. In this case we select each first element of the citation and put it into a data frame.

For the visualization of our sample network, we know that our network will be composed of nodes and edges. Therefore, we have to primarily dissociate our document from our citations cited, by applying specific colors for the representation of the network.
The idea is to illustrate with a graph each ID document as a node in red, as a node master and the corresponding edges (number of volume cited) in blue and see which cases could be clustered together.

For this experiment, we will focus on the html feature.
The html is composed of the following features :
case_cite, parties, docket, court, date, indent

The docket feature represents a number proper to each opinion that is an official court record book which lists all the cases before the court.
The indent feature is a little summary of the verdict of the case in front of the Supreme Court of the United States.
For now, only a few features are interesting, with are the '*case_cite*' and '*ID*' that we extracted from the JSON files. Once the specific strings (corresponding to the citation cited) and corresponding ID's are organized, we can create our network and visualize it in Fig 2.

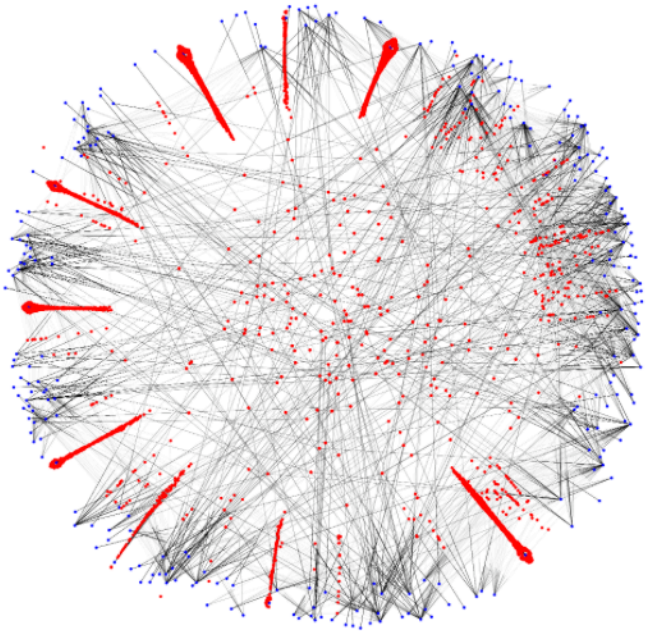

Fig 2. Network of our sample opinions from SCOTUS. Red dots are opinions and blue dots are citations.

From this network, we are able to see some red lines, those represent a cluster of documents that seems to cite the same citation. Indeed, by zooming on those clusters, we are able to see this hypothesis, all those opinions cite a one and unique citation in many cases, this citation can be seen as popular of important. Fig 3.

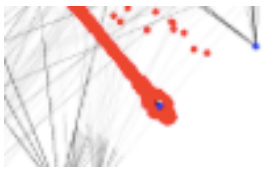

Fig 3. Zoom on part of the network of opinions from SCOTUS, showing a highly cited citation. Red dots are opinions and the blue dot is citation.

One question could pop up being what is our most important case cited. This question can be answered by analyzing the in-degree vertices and the closeness centrality of those vertices.

For our sample, the closeness centrality returns us the most popular case cited. In our case for 5000 documents, those citations where:
- 537, 0.1302
- 543, 0.1232
- 540, 0.0869

The first integer corresponds to the volume of the United States Reports and the second correspond to the closeness centrality value.
For the in-degree vertices, we find the same result as for the closeness centrality. This is because we are having a directed graph. Those returned values, show that for our sample, most case cited are within the volume 537,543 and 540 of the United States Reports.

Once our graph is constructed, we can create a laplacian matrix and calculate the specific eigenvalues and eigenvectors of our laplacian graph matrix. As recall, we eliminate the first eigenvector as this one is equal to zero, and plot our data according to our 2nd and 3rd eigenvector value (Fig.4)

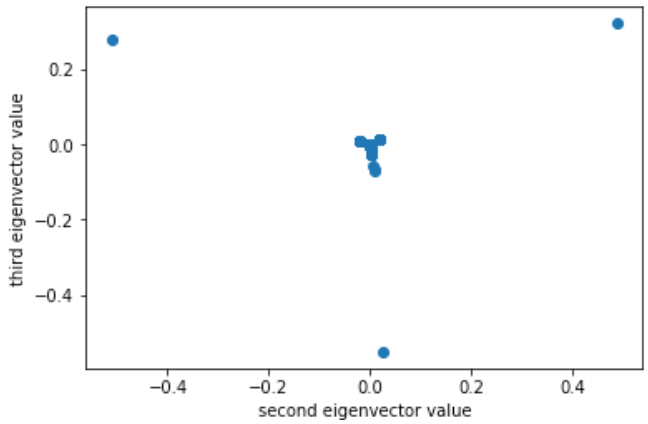

Fig 4. Visualization of our data on the 2nd and 3rd eigenvector.

Already from this figure, we are able to see some distinct clusters that we will have to label. To visualize how many centroids are needed for this clustering, we use the Eclust method to evaluate the smallest sum of squared error (SSE) within our clusters. (Fig.5) (Table.1)

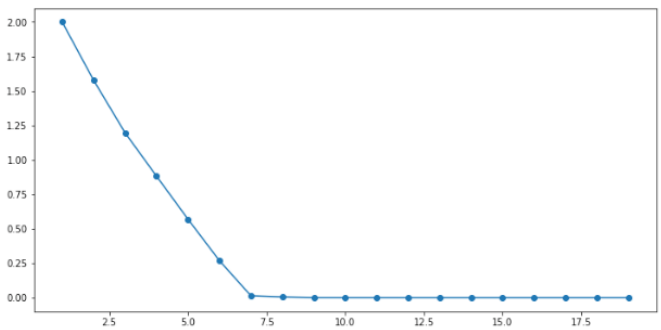

Fig 5. Eclust Method to determine number of clusters.

# of cluster	SSE	# of cluster	SSE
1	1.99	5	0.56
2	1.57	6	0.26
3	1.19	7	0.01
4	0.88	8	0.004

Table 1. SSE compared to the number of clusters.

From this table, we will choose 7 clusters. Now we are able to proceed with K-means on our dataset, and predict our labels for each cluster (Fig.6) and after we would assign each cluster's ID to the dataset (Table 2)

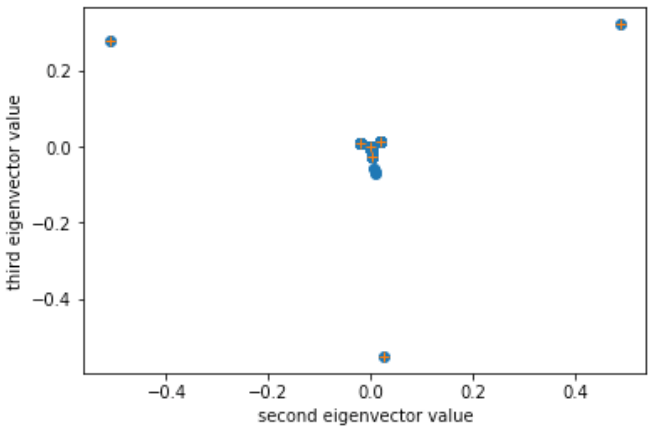

Fig 6. Visualization of our centroids from K-means.

Opinion ID	Case_cite	Cluster_id
143119	544	0
122028	536	0
92451	130, 9, 32	0
134300	540	2
89793	97, 24	0

Table 2. Example of label clustering for the specific opinions.

To conclude this part, we were able to have a proper visualization of the data to give a structured vision of the opinions and their case citations. The main idea was to extract some valuable information

regarding on how those opinions where linked to certain cases, and which cases were most commonly cited. This allowed us to form clusters regarding specific citations and could help us to determine which kind of trial we are having, depending on further sentiment analysis of the case's indent.

4.3 Outcome Prediction & Observation labeling

This part will analyze the feature *indent* which might from a brief look up contain the decision of the specific case. The approach will be to analyze the text contained in the feature *indent* and use it as reference.

To do this, we used the package Stanford openNLP available under Java. Using the R statistical programming language for this part, we had to convert, beforehand, this package to use it in R.

Beforehand, the goal is to clean the data, as converting the text content to lower case, removing numbers and unnecessary punctuation. This will facilitate the sentence extraction from the text.

To do this, we annotate the text indent with sentence and word token annotation, and convert it to an annotated plain text document, with function available in the package.

The next important step is to determine a list of legal words employed by judges to extract the verdict without reading the entire indent feature that can be quite long. Indeed, here the difficulty is to have a complete list of words, in order to extract every verdict for each opinion. The main issue was that according to each opinion, the vocabulary employed to express that verdict changes. For instance, some indent contained the word Certiorari accepted/denied, motion accepted/denied or even judgment accepted/granted/denied. Those are the most common outcomes, but some specifications exist depending on the legal jargon. A research had to be done in order to extract every possible motion.

Once this was performed, we proceeded to the sentence extraction. The method was to skim the annotated plain text document and for each sentence, we would check if one of the words from our reference list is contained into that sentence. If it is the case, we return the entire sentence, and therefore come with an outcome of a verdict. In general, in none of the opinions in our sample, we had an error of multiple outcome for a verdict.

This part was so far successful, to go deeper in the analysis of those outcome, we employed sentiment analysis to estimate a potential label.

The packages used for this occasion are tidytext and sentiword. They will help us to label specific sentiments present in a sentence compared with a reference list of words available in the packages.

So first, the idea was to create a label according to the sentiment analysis of the verdict. A negative outcome would be label -1 and a positive outcome would be label 1. (Fig.7)

['Petitions for writs of prohibition denied.']	140925	petitions for writs of prohibition denied	-1
['C. A. 11th Cir. Certiorari denied. Reported below: 99 Fed. Appx. 881.']	139985	certiorari denied	-1

Fig 7. Abstract of the data frame with the outcome of the indent feature (1st column). 2nd column is the ID of the opinion, the 3rd column corresponds to the outcome of the verdict and the 4th column is the sentiment labeling.

Further, the outcome of a verdict can have positive, negative words or other words corresponding to a specific sentiment. By applying sentiment analysis, we will return the number of words, as a counter, that belong to a specific sentiment compared to the reference list from the package. An example for a verdict outcome will be showed in Fig.8.

"on the whole we are of opinion that the decree of the circuit court dismissing the libel of the captors ought to be affirmed and that the cause should be remanded to the circuit court for further proceedings as between the united states and the claimants"

```
anticipation    2
trust           2
negative        1
positive        1
```

Fig 8. Outcome of our sentiment analysis regarding a specific sentence. The number right of the

sentiments are counters of how many words belong to that class.

Even if overall, the sentiment analysis had good results, some sentences were misclassified. The difficulty of sentiment analysis relies on how the sentence is written and how a normal human being would read into it.
An example can be showed as follow:

"for these reasons this action cannot be maintained and the judgment for the defendant must be affirmed" is **misclassified**

Here, the sentence is turned regarding the defendant part and not the plaintiff. Our labeling is then misclassified.

To conclude on this part, we were able to extract the outcome of the verdict from the indent feature, thanks to sentence token annotation technique. Once extracted, sentiment analysis was performed to label the potential verdict regarding if this is positive or negative outcome. We didn't just stop with only labeling the outcome but we dug a little bit further in a way to see on what kind of sentiments those outcomes are based. Sentiment analysis showed us some great results but in some cases, we saw some limitations as in the misclassification of some judgments.

5. Further Experiments

Here we focused more specifically on the cluster analysis. Indeed, compared to the previous experiment that clustered opinions based on only the volume of the United States Report, here we clustered our opinions based on the volume and the starting page of the case. Following the same procedure as above, we visualize our data within a network. As our data sample is pretty small compared to the actual number of opinions within SCOTUS, there is a small chance that actually opinions are citing the same case.

From our closeness centrality measurement, we are now able to extract the most cited cases from our data sample which are:

- 544 US 971 : Byrd vs United States (2005), was cited 5 times in our data sample. This number is obtained by analyzing the degree vertices.
- 537 US 1149 : In Re Thomas, Ante, P.810 (2003) was cited 7 times.
- 543 US 872: Smith, Sheriff, Smith County, Texas vs Kinney And (2004) was cited 3 times.

The further idea would be to implement this method on the entire dataset using AWS to lower the computational cost and get the most cited cases over the entire SCOTUS jurisdiction.

6. Conclusion

The project enabled us to give an interesting insight on law data. Once the data was preprocessed, we were able to have the first outcomes of our experiments. These experiments consisted in clustering and labeling opinion cases based on their verdict by using sentiment analysis.

One of the main obstacles we had to overcome and found most difficult was the information extraction from the opinion files, as they differ on each jurisdiction and on the inconsistency of the data. Considerable amount of time was spent on how to properly extract this information and understanding law data. We hereby see that the bases of good analysis and research into data could be improved with more accurate feature information.

Indeed, we could improve the dataset by creating a feature that states the winner of the trial and the final outcome, for a better data overview. Based on our ground work, the second part of the project was organized to go further into depth looking for a potential correlation between the verdict and the political affiliation of judges, or use sentiment analysis in order to connect it to new cases, based on the jurisdiction and what kind of case judges are facing (bankruptcy, crime, tax evasion...). A deeper look into the clusters would maybe have given a better representation of why they are labeled that way.

The input of professionals of the law, regarding what they which to seek through this data, would have led to a different approach. Indeed, we started with the SCOTUS jurisdiction, which is the final stage of the case. It might have been interesting to start with First Circuit of Appeals to determine what kind of case establish some text mining to label a specific topic. We could have combined

these outcomes with the in parallel study on SCOTUS.

To highlight the main point, the possibilities are without limit, a better comprehension and guidance before handling this data might have accelerated the process of valuable ground work being done. Meanwhile, our outcomes are encouraging enough for providing a possible future and expansion of this project.

7. Personal input

My input in this project was to work on clustering analysis and outcome extraction with Antoine Gargot, one of my team peers. Further, I expanded my experiments regarding clustering on the opinion volume and page from the United States Report.

8. References

[1] https://www.courtlistener.com/api/bulk-info/

[2] https://www.courtlistener.com/api/jurisdictions/

[3] https://github.com/idc9/law-net

[4] https://www.youtube.com/watch?v=AP7_godzwVI

[5] https://www.courtlistener.com