# Authorship Attribution on SCOTUS Opinions

Morgan Cassels
SENG 474
April 3, 2019

# The dataset

- Supreme Court of the United States (SCOTUS) opinions
- ~35,000 opinions, 96 different judges, from 1789-2017
- Data set includes, for each opinion:
  - Author
  - Body text
  - Direction: conservative or liberal
  - Type: majority, concurring, first dissent, or second dissent
  - Number of justices voting in the majority and minority

Data from: https://www.kaggle.com/gqfiddler/scotus-opinions

# Authorship attribution

- Determine who, among a set of known authors, wrote a document
- Used for analysis of historical texts, prosecution of cybercrime [1], etc.
- Subdomain of text classification with distinct characteristics:
  - Content/topic is not relevant
  - Stylistic features must be used

# Constraining the problem

- Remove all data other than author and the body text of the opinion — only data is the text
- Remove very brief opinions (<3000 characters)
- Use top five most prolific authors only
- 5th most prolific wrote 815 opinions — use 815 from each to make total dataset of 4075 opinions

# Approaches in the literature

- Many approaches involve using labelled data to train a classifier (supervised learning)
- There is less literature on unsupervised approaches:
  - Hacohen-Kerner and Margaliot [1]:
    - Corpus: Responsa (answers to religious questions) written by 5 Jewish rabbis
    - Term frequency
    - 2 non-hierarchical methods: K-means and Expectation Maximization
  - Mingzhe and Minghu [2]:
    - Corpus: the novels of 5 famous Chinese authors
    - Bigrams and punctuation usage
    - K-means and Hierarchical clustering

# My approach to classification

- Unsupervised — Clustering where k = #authors
- The “correct” author of a cluster = author who wrote the most documents clustered to that cluster
- Tried 2 types of clustering:
  - K-means
  - Hierarchical clustering with Euclidean distance and Ward linkage (minimizes variance of clusters being merged)
- Performed clustering between 2, 3, 4, and 5 authors

# Measuring classification accuracy

Example classification result when k=2:

| Cluster index | # opinions written by Harlan | # opinions written by Douglas |
|---|---|---|
| 0 | 266 | 812 |
| 1 | 549 | 3 |

Since most of the opinions in cluster 0 were written by Douglas, we say that Douglas is the "correct" author for cluster 0; similarly Harlan is the "correct" author for cluster 1.

Confusion matrix:

| | Labelled 0 | Labelled 1 |
|---|---|---|
| Should be 0 (Douglas) | 812 | 266 |
| Should be 1 (Harlan) | 3 | 549 |

| Cluster Index | Correct | Noise | Silence | Precision | Recall | F-Measure |
|---|---|---|---|---|---|---|
| 0 | 812 | 266 | 3 | 0.753 | 0.996 | 0.858 |
| 1 | 549 | 3 | 266 | 0.995 | 0.674 | 0.803 |

**Total percent correct: 83.5%**

**Improvement rate = percent correct - baseline = (0.835 - (1/2) )*100% = 33.5%**

# Creating document vectors

- Methods of tokenizing
- Named entity recognition
- Consideration of stop (function) words (e.g. ‘and’,  ‘is’, ‘too’)
- TF or TF-IDF
- Word vectors —  likely not appropriate (topic is not related to authorship)
- Vocabulary diversity —  likely not appropriate (all authors are highly educated judges)

# Tokenizing

sentence = "A. Smith shouldn't have eaten Mike's sandwich (but he did)."

Naive tokenizing:
['A.', 'Smith', "shouldn't", 'have', 'eaten', "Mike's", 'sandwich', '(but', 'he', 'did).']

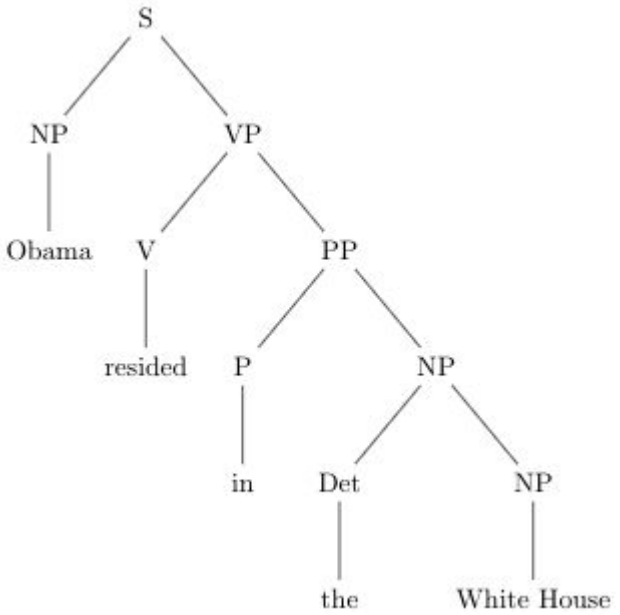
Tokenizing with nltk:
['A.', 'Smith', 'should', "n't", 'have', 'eaten', 'Mike', "'s", 'sandwich', '(', 'but', 'he', 'did', ')', '.']

# Named entity recognition

```
>>> sentence = "Obama resided in the White House"

>>> nltk.ne_chunk(nltk.pos_tag(sentence.split()))
```

```
Tree('S', [Tree('PERSON', [('Obama', 'NNP')]), ('resided', 'VBD'),
('in', 'IN'), ('the', 'DT'), Tree('FACILITY', [('White', 'NNP'), ('House',
'NNP')])])
```



# Term Frequency - Inverse Document Frequency

TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)

IDF(t) = log_e(Total number of documents / Number of documents with term t in it)

- TF considers each term equally
- TF-IDF gives more weight to rarer terms

# Different parameters for document vectors

- With and without Named Entity (NE) removal
- TF vs. TF-IDF
- With stop words, without stop words, only using stop words
- With and without punctuation
- How many tokens to use i.e. how long should the document vector be?
    - All words (68,211)
    - 500, 1000, 2000, 2500, 3000, etc.  most frequent

# Parameters for best results

- Keep stop words and punctuation
- 2000 most frequent tokens
- TF; not TF-IDF
- Removing or keeping Named Entities does not make a significant difference

# Best clustering results

- Achieved with Hierarchical clustering; K-means was slightly less accurate
- K = 2, 3, and 4 results varied based on which authors were considered — some authors have a more distinct writing style

The following are averages across all clusters and all combinations:

| k | percent correct | improvement rate |
|---|---|---|
| 2 | 77.4% | 27.4% |
| 3 | 68.8% | 35.3% |
| 4 | 63.4% | 38.4% |
| 5 | 60.0% | 40.0% |

# General findings and future work

**Findings**

- Stop words and punctuation are meaningful for authorship attribution
- There is a threshold after which considering more words *reduces* classification accuracy
- Named Entity removal is not necessary — likely because these words are already less common
- Hierarchical clustering produces better results than K-means ([3] found the same)
- The accuracy of 2-way clustering is highly variable depending on which two authors are compared

**Future Directions**

- n-grams (sequences of n characters or words)
- consideration of sentence length
- n-grams using punctuation to consider sentence structures

# References

[1] S. Seifollahi *et al*, "Optimization Based Clustering Algorithms for Authorship Analysis of Phishing Emails," *Neural Processing Letters*, vol. 46, *(2)*, pp. 411-425, 2017.

[2] Y. HaCohen-Kerner and O. Margaliot, "Authorship Attribution of Responsa using Clustering" *Cybernetics and Systems*, vol. 45, *(6)*, pp. 530-545, 2014.

[3] J. Mingzhe and J. Minghu, "Text clustering on authorship attribution based on the features of punctuations usage," in 2012, . DOI: 10.1109/ICoSP.2012.6492012.