Stylometric Analysis of Raw Tweets Using Scikit-Learn

William
GA-SEA-DAT2

What is Stylometry?

- **The Problem:** Identifying authorship of text through analysis of *writer invariants.*

- **Writer Invariants** are *features* of text passages from the same author that...
    - Are statistically unique to the same author
    - Remain largely unchanged throughout text

- **Features** that may help identify invariants:
    - **Numeric:** *Average sentence length, syllables, etc*
    - **Synonyms:** *Word choice, use of contractions, etc*
    - **Punctuation:** *Comma, semicolon, or hyphen use*

The Question(s)

- **In the absence of usernames, what *features* can we best identify tweet authorship with?**

- **Are there features characteristic to tweets that make them easier to attribute?**

- **Hypothesis:** Despite the 140-character limit, tweets may have other useful invariants:

  - *Frequency of **link usage**, which **domains***

  - *If added text is placed before/after links*

  - *Hashtag usage, response to other users*

The Data

- **Source:** Acquire tweets in CSV form for model. I have chosen to use feeds of all US senators.

- **Quantity:** Current research indicates that *five-thousand words* is point of diminishing returns. Given Twitter's 140 character limit, I suspect *five-hundred* tweets per senator will suffice.

- **Wrangling:** The CSV files of the Tweets must be organized in a way that makes feature engineering easy to do. I may need to revist this step based on the features I hope to use.

```python
def __main__():
    from twython import Twython
    print 'test'

    twitter = Twython(                                )

    from pprint import pprint
    #pprint(twitter)
    search_result = twitter.search(q = 'stylometry')
    #pprint(search_result)
    for status in search_result ['statuses']:
        user = status['user']['screen_name'].encode('utf-8')
        text = status['text'].encode('utf-8')
        print user, ":", text
        print
__main__()
```

```
test
ml_applications : Annotating A Book Using Syntactic Stylometry

kyrre__ : stylometry gone wrong https://t.co/uCZ36N5ude

RobertVinluan : RT @icebergdotcool: New Post!

▯:  Is this leaked script real? Gwern uses stylometry and probability to find out.

https://t.co/6UuSMgD8ij

icebergdotcool : New Post!

▯:  Is this leaked script real? Gwern uses stylometry and probability to find out.

https://t.co/6UuSMgD8ij

JuolaAssociates : Software could expose the identity of hackers -- Defense Systems https://t.co/lY9FOPyWlm

jeaner823 : Software could expose the identity of hackers https://t.co/KZOIL9OfNZ

silly_phrases : nonemergent stylometry
```

The Modeling

- **Logistic Regression:** Data is not numerically continuous, quantities and yes/no for features.

- **First:** Analyze twitter feed by chosen features.

- **Next:** Choose which features best distinguish one feed from another, much iteration...

- **Final:** Gather a subset of new tweets from previously selected feeds (ten or so), remove the user handles, and use model to attribute.

Example: Syllable Count

- To be used in Jupyter:

```python
773
774    def syllables(word):
775    count = 0
776    vowels = 'aeiouy'
777    word = word.lower().strip(".:;?!")
778    if word[0] in vowels:
779        count +=1
780    for index in range(1,len(word)):
781        if word[index] in vowels and word[index-1] not in vowels:
782            count +=1
783    if word.endswith('e'):
784        count -= 1
785    if word.endswith('le'):
786        count+=1
787    if count == 0:
788        count +=1
789    return count
```

What Now?

- **Coding:** Python code such as what you saw will be ported into a Jupyter notebook.

- **Data Acquisition:** Scraping tools allow for more tweets, API via Twython is kind of limited…

- **Build the Model:** Of all notebooks we've run, this project is most similar to the Bank analysis

- **Gather New Tweets:** These will have names removed, and my hope is to build a model that will attribute tweet source to the right senator.