Python, Sklearn and some Summarization

COMP 599

22nd September

Introduction to Python

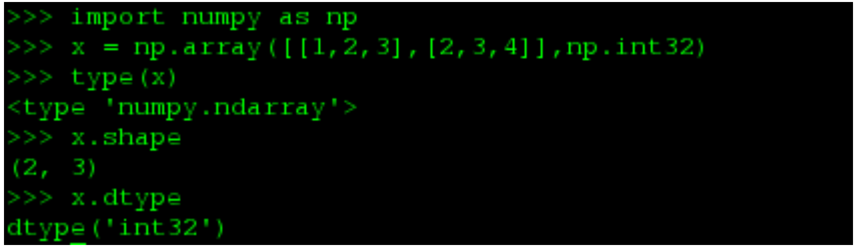
- Open a prompt with Python. (Make sure it is version 2.7)
- Example code :
  - look for : indentation, for, if, else-if constructs, methods, compulsory and optional variables
- Some common commands for text used:
  - split, join, substring search

Classes in Python

Example code

Numpy

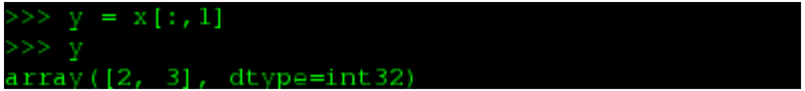
- Python scientific computing package.
- Install with a Python package installer.
- N-dimensional arrays in numpy :
  - Example of array creation :



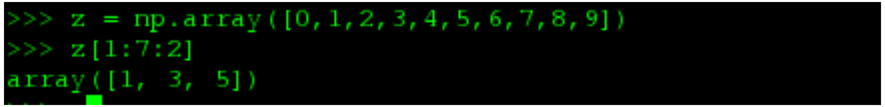
```
>>> import numpy as np
>>> x = np.array([[1,2,3],[2,3,4]],np.int32)
>>> type(x)
<type 'numpy.ndarray'>
>>> x.shape
(2, 3)
>>> x.dtype
dtype('int32')
```

Array slicing

- Generate views of the data.
- Slice object - **start : stop : step**



```
>>> y = x[:,1]
>>> y
array([2, 3], dtype=int32)
```



```
>>> z = np.array([0,1,2,3,4,5,6,7,8,9])
>>> z[1:7:2]
array([1, 3, 5])
```

Scikit learn

- Machine Learning package for Python.
- Example code (Linear regression).

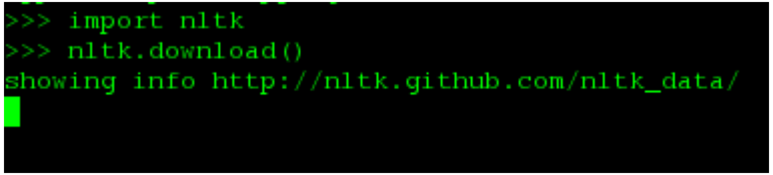
NLTK

- What's NLTK?
  - Natural Language ToolKit
- What does it contain?
  - Stemmers, lemmatizers, parsers with a bunch of corpora

NLTK data

Downloading the data :

Open python and type the following commands :



```
>>> import nltk
>>> nltk.download()
showing info http://nltk.github.com/nltk_data/
```

Contents

- Natural Language Generation
- Summarization
  - Extractive
  - Abstractive
- ROUGE evaluation metric
- Stylistics - Formality, subjectivity
- Tweet generation
- Indicative tweets using articles
  - Data
  - Results
  - Interaction with Formality
- Conclusion

Natural Language Generation

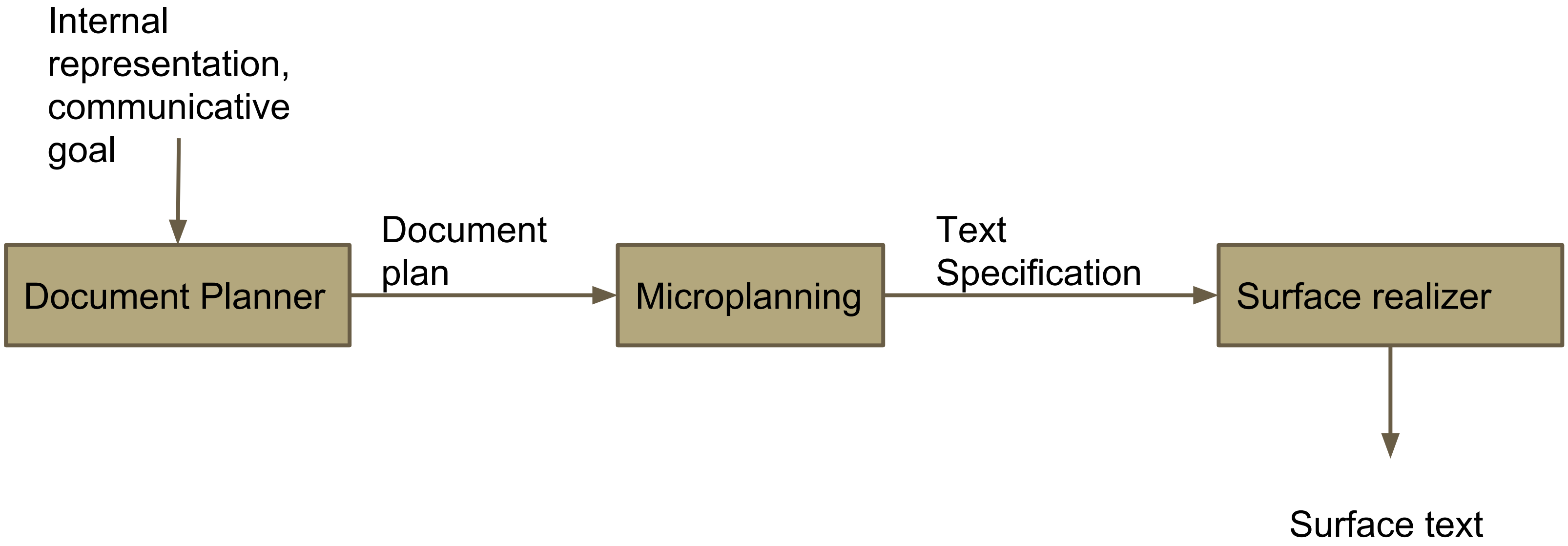
- Generating understandable text from machine representation of information

- One of the first NLG systems : Weather information system WeatherReporter

- Natural Language Understanding vs Natural Language Generation : hyothesis vs choice

NLG system Structure

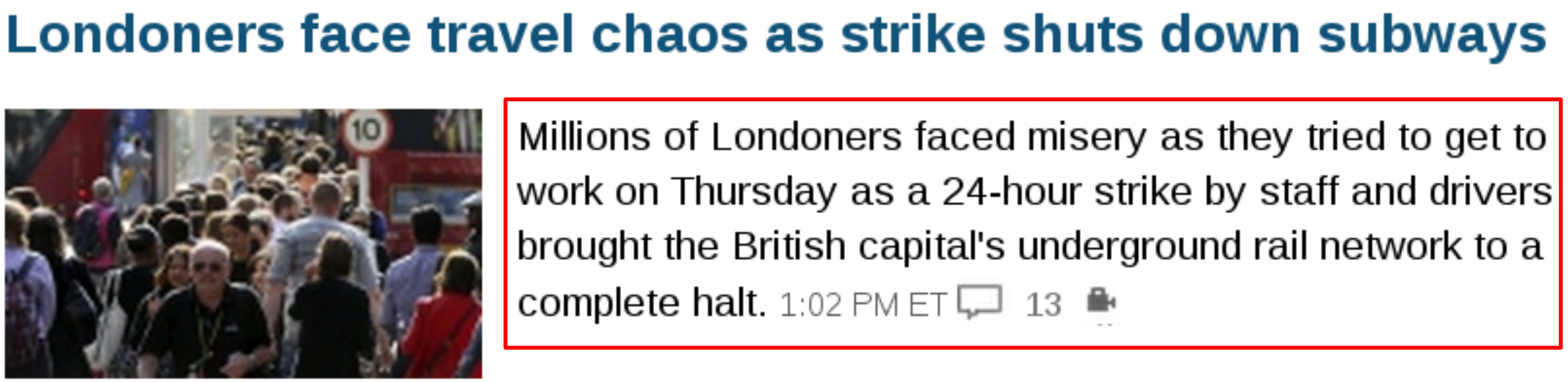
- Broad structure :



Summarization

- Automatic summarization techniques
  - process of reducing text document
  - retain important information from source

- Two main approaches :
  - Extractive
  - Abstractive

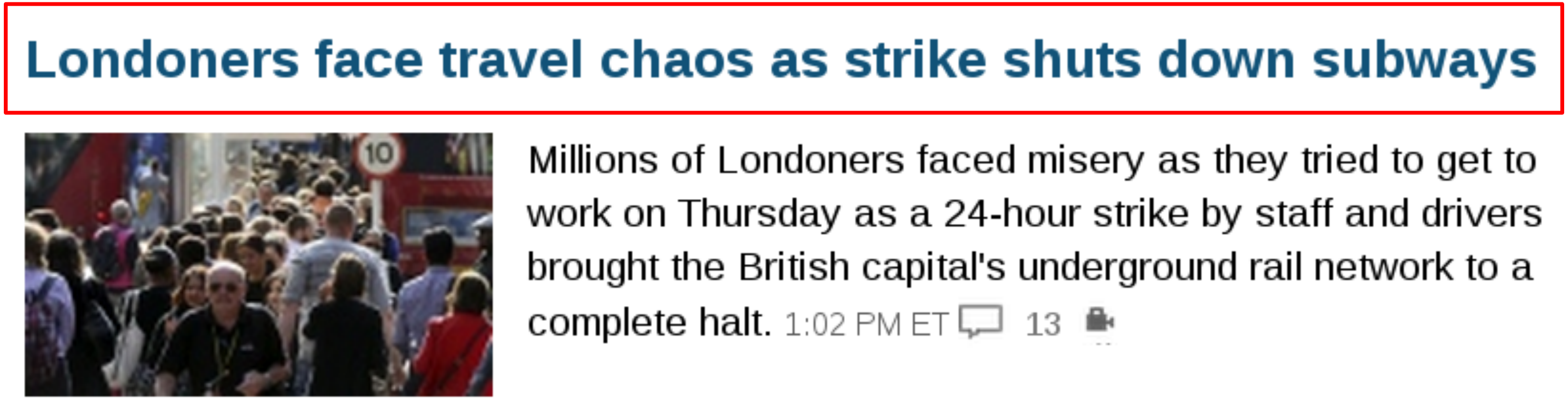


Londoners face travel chaos as strike shuts down subways

Millions of Londoners faced misery as they tried to get to work on Thursday as a 24-hour strike by staff and drivers brought the British capital's underground rail network to a complete halt. 1:02 PM ET 💬 13 🎥

Extractive summarization

- Extract key sentences or paragraphs, piece together
- Relatively simple, retains key information
- Drawbacks :
    - summary is disconnected and incoherent
    - inconcise
    - sometimes misleading
- How to overcome this? Use NLG techniques, smoothe extracted sentences to generate readable summaries



Londoners face travel chaos as strike shuts down subways

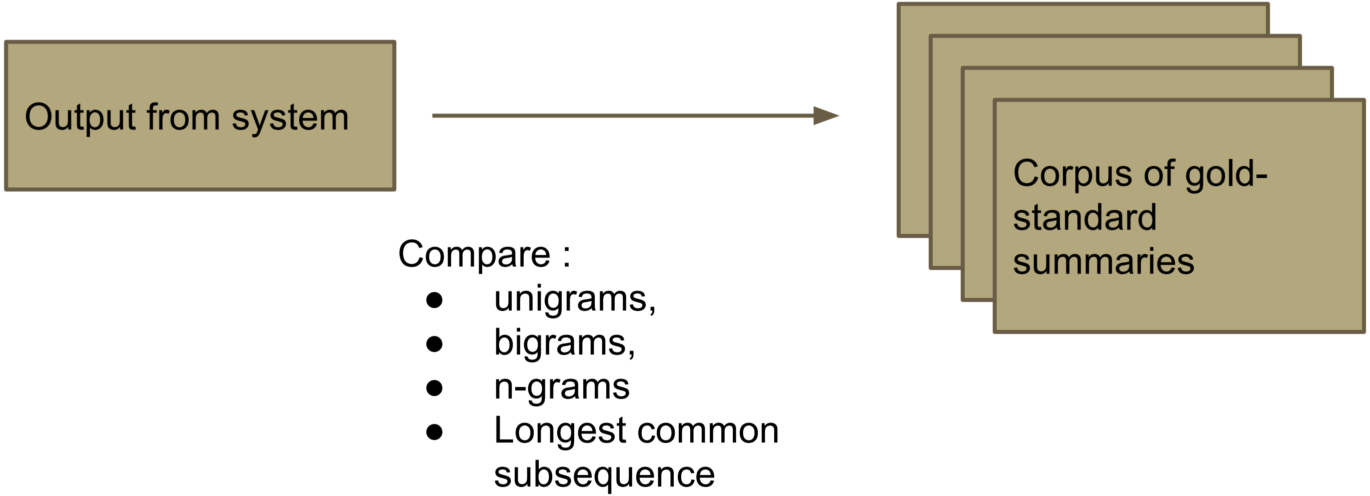
Millions of Londoners faced misery as they tried to get to work on Thursday as a 24-hour strike by staff and drivers brought the British capital's underground rail network to a complete halt. 1:02 PM ET 💬 13 🎥

Abstractive Summarization

- Extract information from text, generate novel sentences to represent it in concise form.
- Usually requires world knowledge, much harder problem
- Summaries are expected to be more coherent and concise than extractive summaries.

Evaluation : ROUGE-1,2,L scores

- Recall Oriented Understudy for Gisting Evaluation
- Used for automatic summarization and machine translation
- 1 - unigram, 2 - bigram matching, n - n gram matching, L - Longest common subsequence match.
- Works best with a range of model human summaries



Compare :

- unigrams,
- bigrams,
- n-grams
- Longest common subsequence

Stylistics

- Information that can be extracted from the text, that is not related to meaning of the text
- Applications
  - Authorship attribution
  - Semantic Analysis
  - Personality Typing
- Stylistic features
  - part-of-speech
  - function words
  - textual statistics - word & sentence length

Formality

- Is also a stylistic feature, associated with interpersonal status, social standing

  get, acquire, snag, obtain, appropriate

- Studies for obtaining lexicons - Julian Brooke, recent paper from NAACL [2],[3]
- Applications in text summarization, machine translation, classification etc.

Subjectivity

- Subjectivity lexicon - words that might indicate opinion in text.
- Example :

adore, agree, scary, selective

- Obtained using manual annotation and then using a polarity classifier.
- Words classified as strongly subjective and weakly subjective.

Stylistic features in NLG

- Can be used as parameters in generation
- Dimensions that have been used - colloquialism, politeness, naturalness [1]
- Use style scores as parameters while generating further text.

Tweet generation

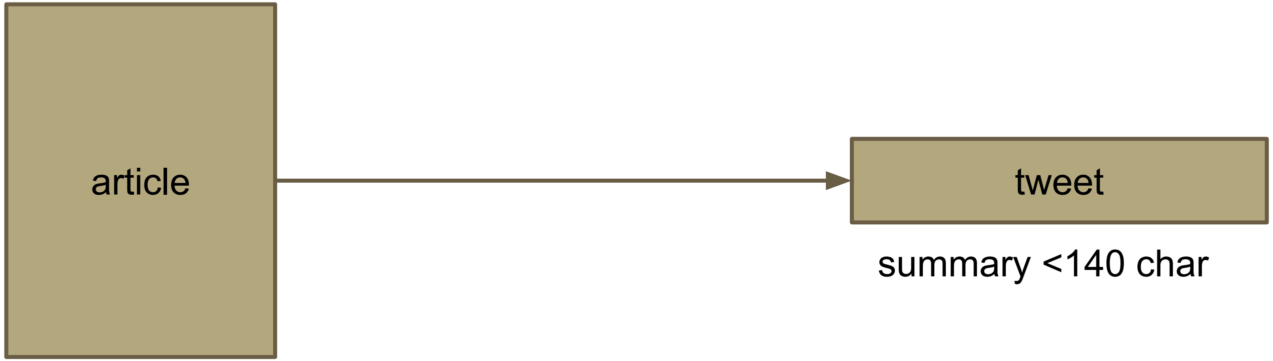
- Applications in advertisements, event summarization.

- Has been talked about a little
  - use existing summarization techniques to generate tweets
  - suggested : use documents from local public works office for updates

Idea

- Indicative tweets - ones that contain link to another article

- Intuitive to think of it as extractive summarization problem



Earlier attempt

- Study compared various summarization algorithms to generate tweets. [4]

- Used ROUGE and user evaluations. For ROUGE, human written reference tweet taken as gold standard.

- Drawbacks :
    - ROUGE in this case does not make sense.
    - Examples of tweets generated not satisfactory

Data

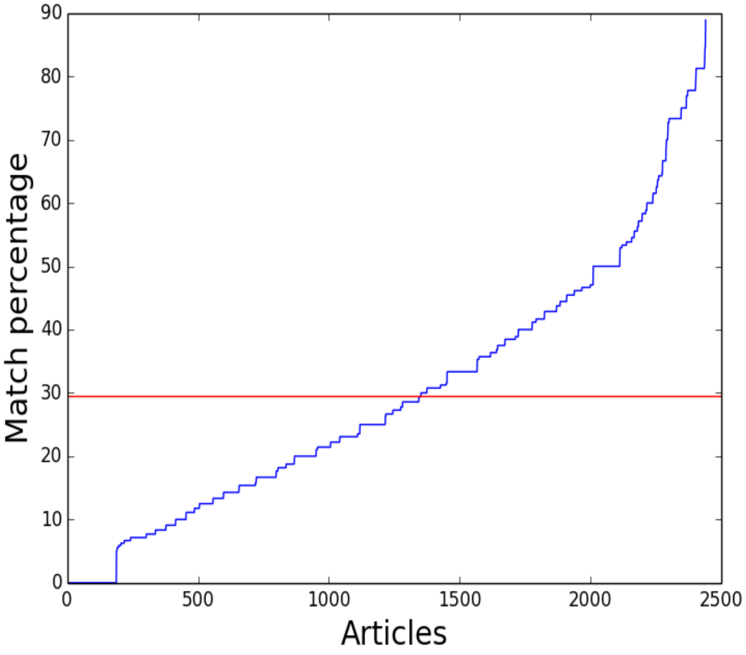
- Tweets from hashtags
- Extract articles from urls connected.
- Data cleaning - images, videos, advertisements, other languages

Politics	**Science & Technology**
#apec2014 #G20 #oscarpistorius	#rosetta #lollipop #mangalayan
Events	**Films and Pop culture**
#haiyan #memorialday #ottawashootings	#TaylorSwift #theforceawakens #johnoliver
International	**Sports**
#berlinwall #ebola #erdogan	#ausvssa #playingitmyway #nycmarathon

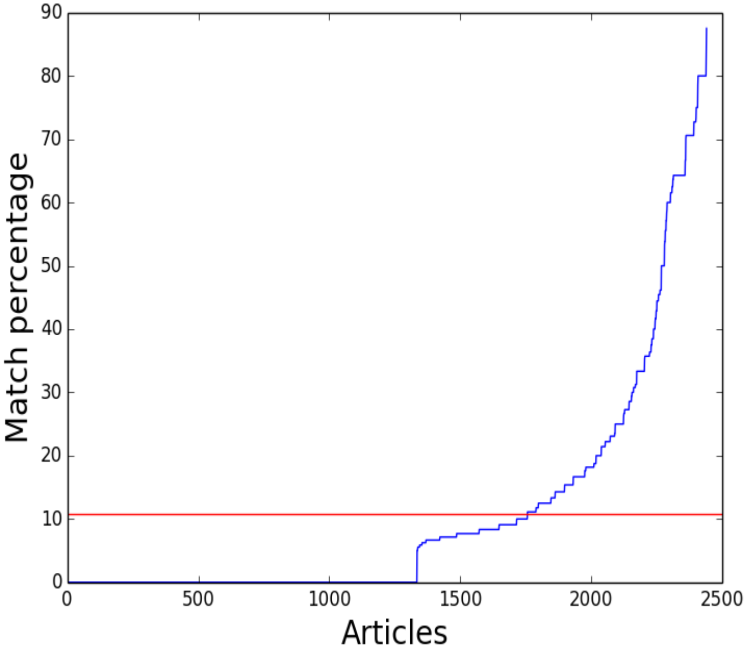
Direction of analyses

- Calculate scores of overlap in tweet & article
- Scores give the degree to which the tweet can be extracted using extractive summarization
- ROUGE-inspired unigram, bigram and LCS matching scores for article-tweet pairs

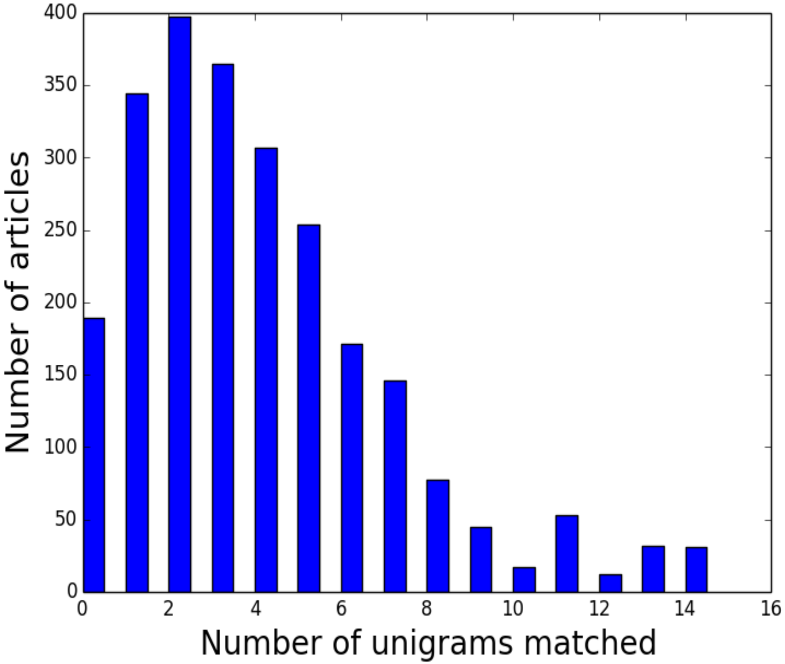
ROUGE inspired scores

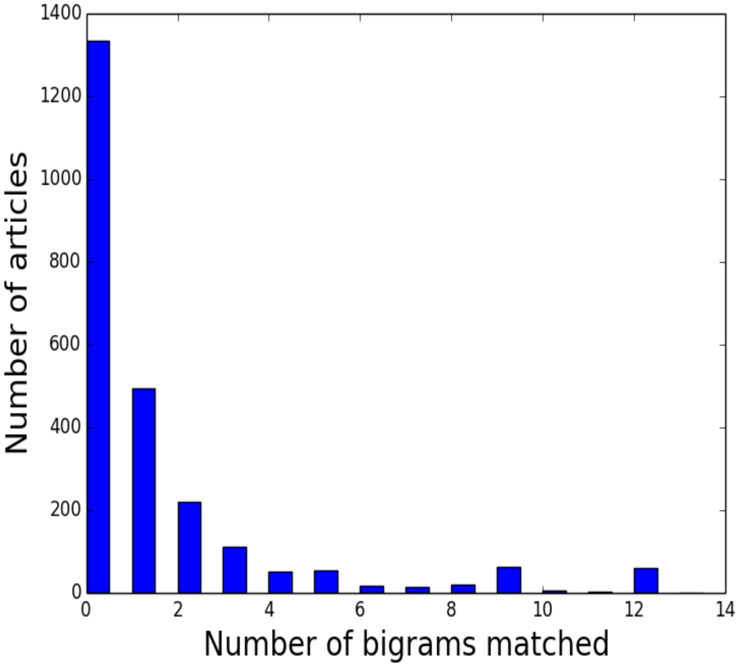


Unigram matching

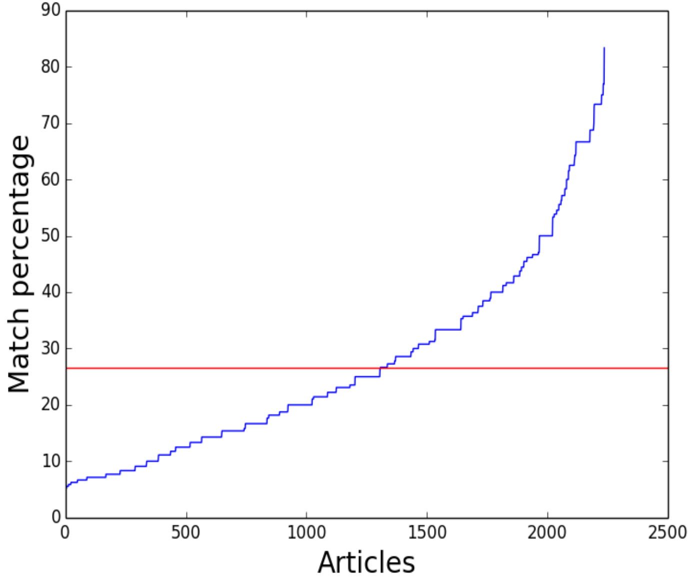


Bigram matching

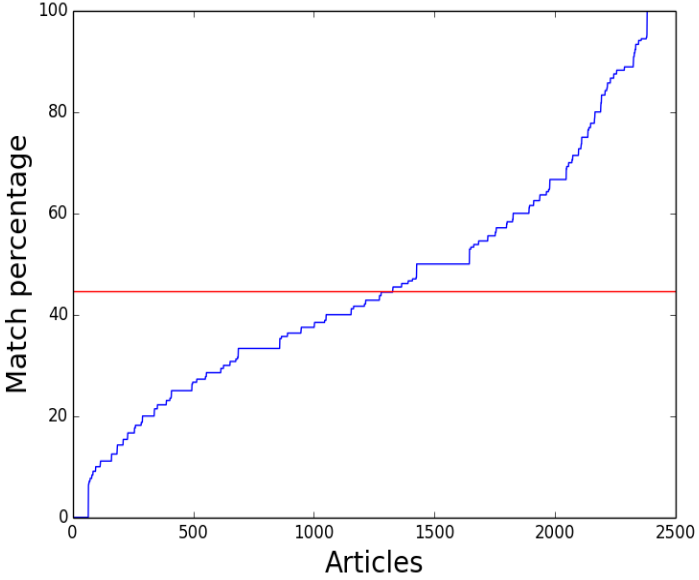

Number of articles
400
350
300
250
200
150
100
50
0
0
2
4
6
8
10
12
14
16
Number of unigrams matched


Number of articles
1400
1200
1000
800
600
400
200
0
0
2
4
6
8
10
12
14
Number of bigrams matched

ROUGE scores



Unigram matching in window = 3 sentences



LCS matching

Interaction with formality

- Formality of articles and averaged over hashtags using lexicon :

Lowest	Highest
#theforceawakens #TaylorSwift #winteriscoming	#KevinVickers #erdogan #apec

- Correlate formality of articles with degree of extraction represented by LCS : Pearson coefficient of 0.41 with p-value of 7.08e-66.
- More formal the article, the more chances that the tweet can be extracted.

Conclusion, next steps

- Results show tweets cannot be generated using extractive summarization

- Use intent - model purpose or intent of tweets.
    - advertisement, opinion, support a cause etc.

- Information on the actual contents of the tweets - why they are not in common with the tweets.

References

[1] Dethlefs, Nina, et al. "Cluster-based Prediction of User Ratings for Stylistic Surface Realisation." *EACL 2014* (2014): 702.

[2] Brooke, Julian, Tong Wang, and Graeme Hirst. "Inducing lexicons of formality from corpora." *Methods for the automatic acquisition of Language Resources and their evaluation methods* (2010): 23.

[3]Pavlick, Ellie, and Ani Nenkova. "Inducing Lexical Style Properties for Paraphrase and Genre Differentiation."

[4] Lloret, Elena, and Manuel Palomar. "Towards automatic tweet generation: A comparative study from the text summarization perspective in the journalism genre." *Expert Systems with Applications* 40.16 (2013): 6624-6630.