Growing Instability: Classifying Crisis Reports

Kim Vuong, RedPixie

The Challenge

Problem

To build a model that can determine the topics for future documents so that they can be classified and used to detect signs of growing instability.

Data

News articles and documents that relate to a humanitarian crisis.

Main Challenges

Dealing with multi-label classification of complex and sometimes ambiguous articles

Using historical documents to predict themes of future documents

Working with text data, or NLP (Natural Language Processing), which is one of the most challenging and interesting fields of Artificial Intelligence.

The Dataset

News articles

Training data: 1.6 million documents

Documents of different lengths

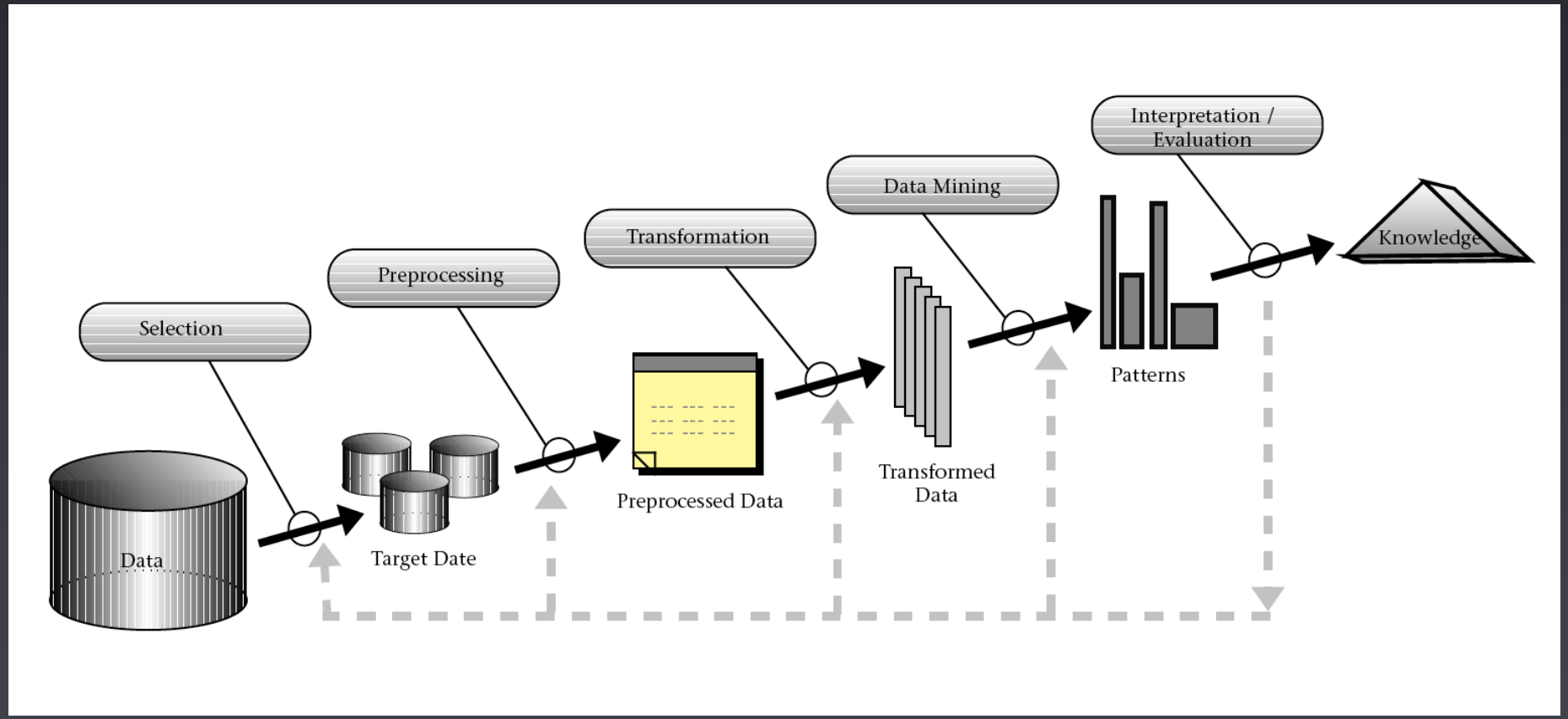
160 topics of interest plus others

Documents published between 2001 and 2014

BodyText	Topics	WebPubDate
The next president of the United States will face real and serious national security challenges on a multitude of fronts, with al-Qaida at the top of the list. Nearly seven years after 9/11, its media outreach programme broadcasts messages on the airwaves and the internet, attempting to radicalize unaffiliated sympathisers into violent action.	['pakistan', 'world', 'alqaida', 'middleeast', 'usnews', 'debate']	29/05/2008

The Process

Data Science Process – Knowledge Discovery in Databases (KDD)



Fayyad, Piatetsky-Shapiro, Smyth, "From Data Mining to Knowledge Discovery: An Overview", in Fayyad, Piatetsky-Shapiro, Smyth, Uthurusamy, *Advances in Knowledge Discovery and Data Mining*, AAAI Press / The MIT Press, Menlo Park, CA, 1996, pp.1-34

Data Selection

Tools used to build solution

python™

spaCy

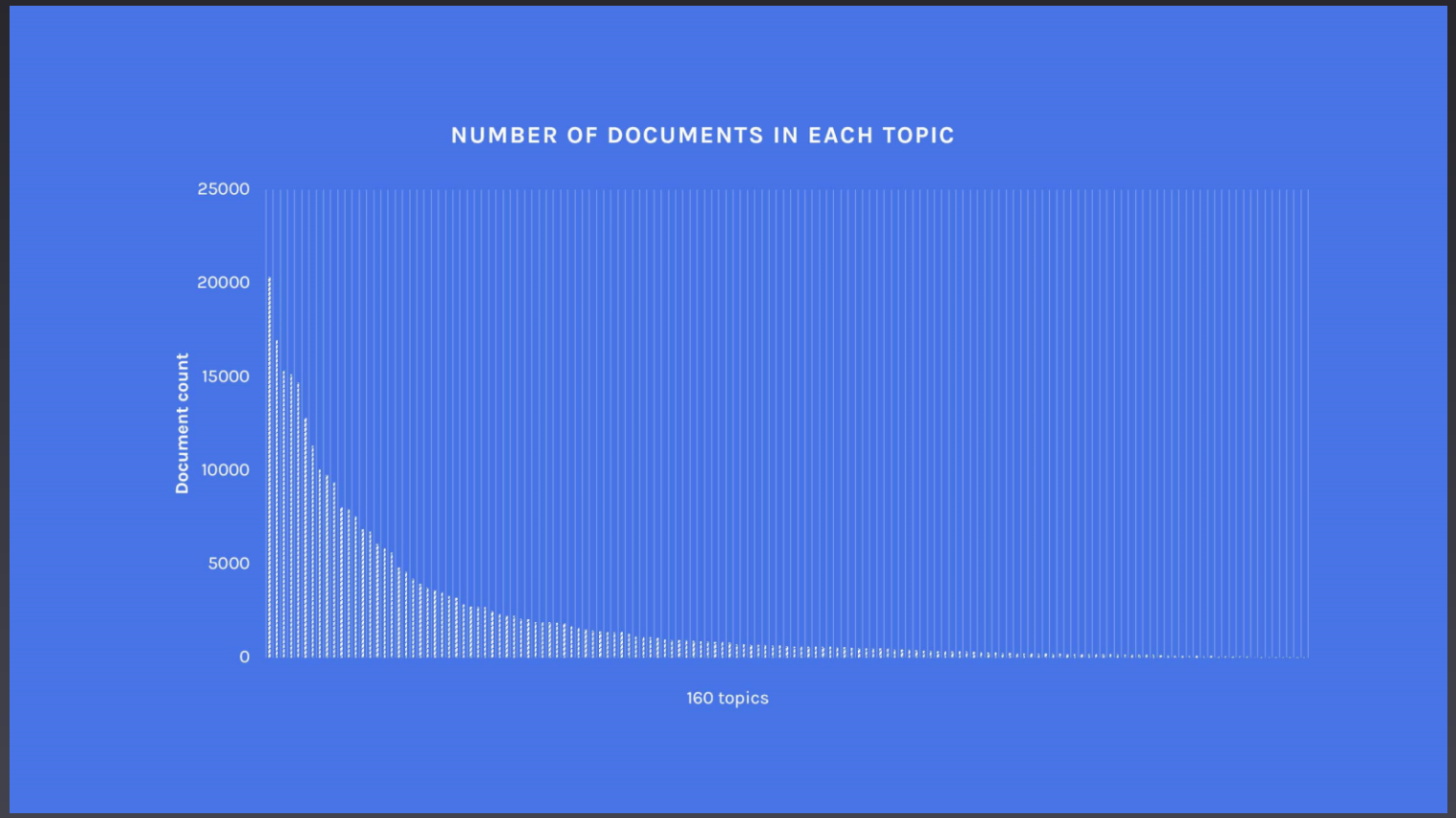
Explore the data

200,000 documents containing topics of interest

146 topics present in the data

14 topics with no documents

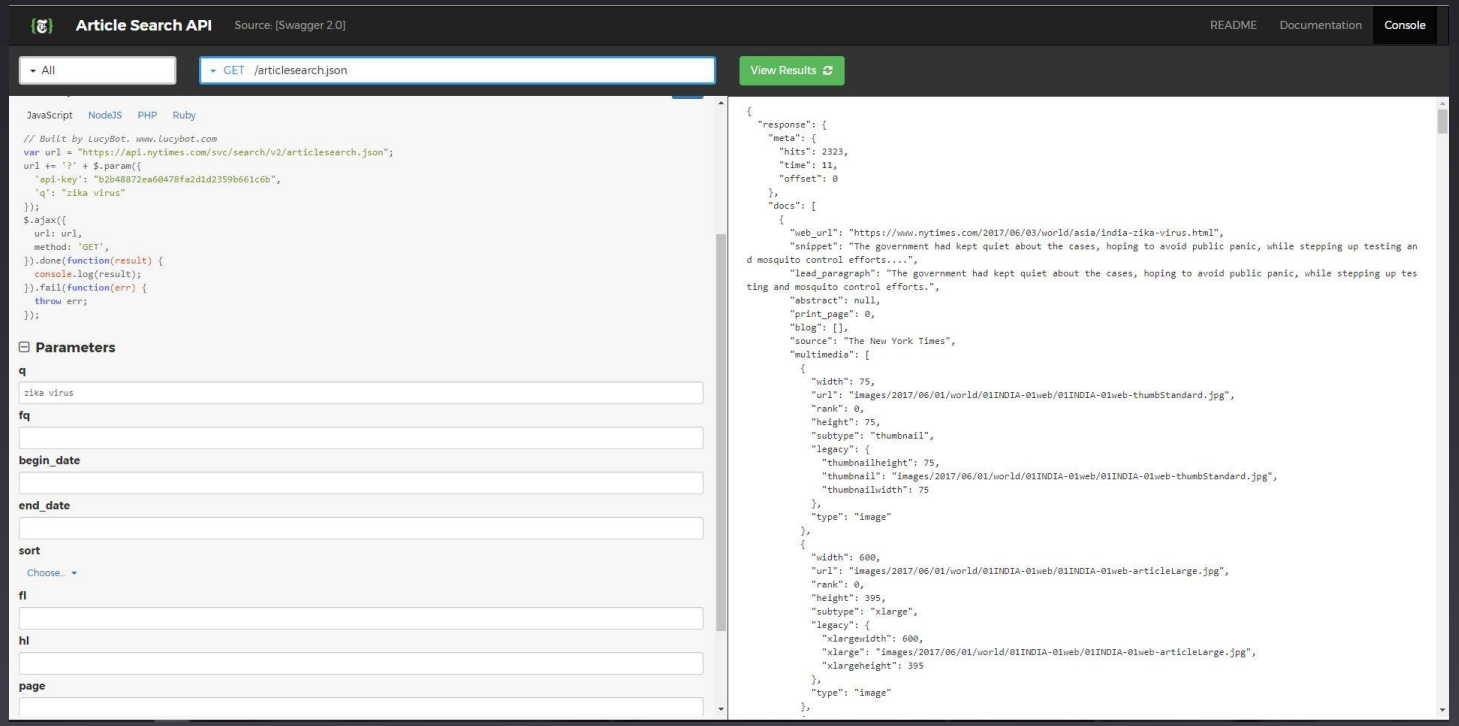
Unequal representation of topics



Naively inferring topics

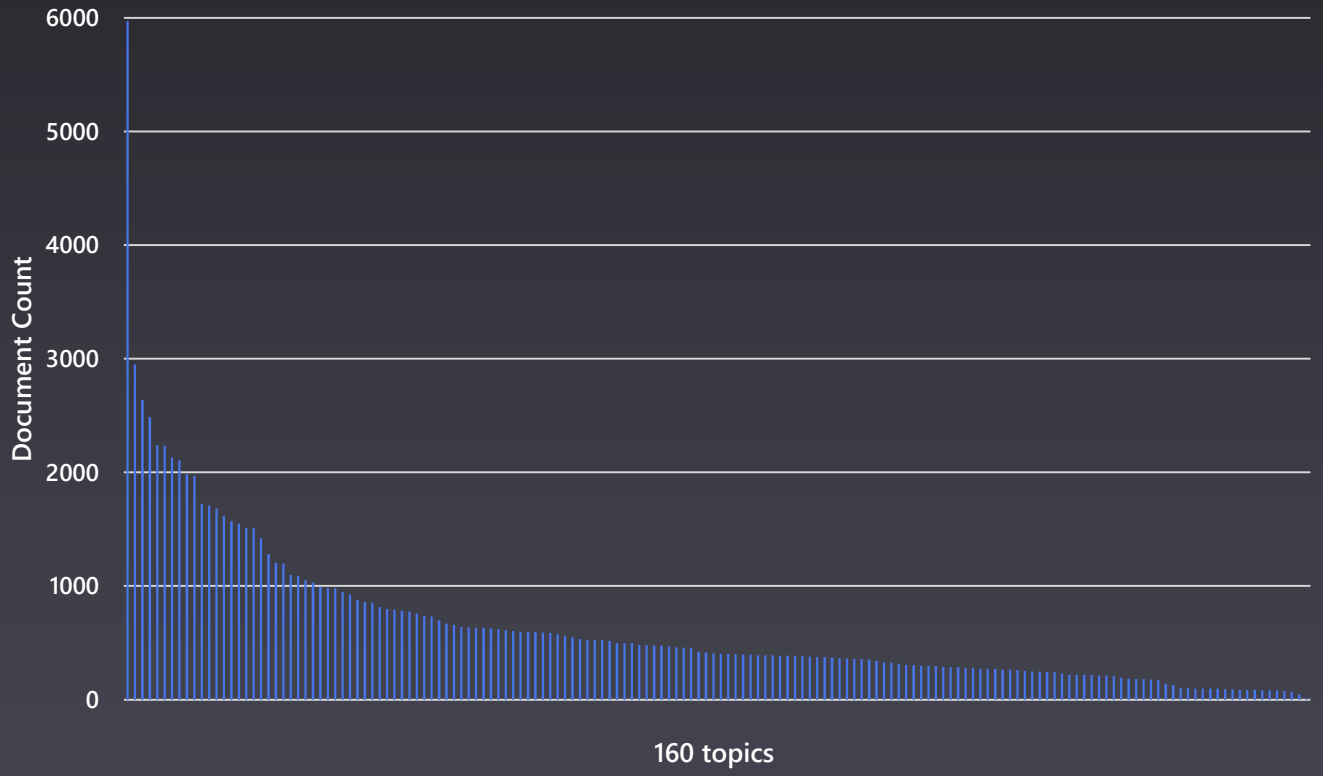
```
25 import pandas as pd
26
27 f = 'trainingdata.csv'
28
29 #formatting
30 df = pd.read_csv(f, encoding='latin')
31 df['strTopics'] = df['Topics']
32 df['Topics'] = df['Topics'].apply(eval)
33 df['BodyText'] = df['BodyText'].fillna('')
34 df['strTopics'] = df['strTopics'].astype(str)
35
36 #function to label documents with new topics
37 def check_contains_label(df):
38     for index, row in df.iterrows():
39         if 'activism' in row['BodyText']:
40                 row['Topics'].append('activism')
41
42 check_contains_label(df)
43
44 df.iloc[264918]
45
46 out[18]:
47 Unnamed: 0                                                  264918
48 BodyText      Sue George is the editor of the Guardians Inte...
49 Topics                          [journalismcompetition, activism]
50 WebPubDate                                           2011-04-21
51 strTopics                ['journalismcompetition', 'activism']
52 Name: 264918, dtype: object
53
54 df2[df2['BodyText'].str.len() < 500]['BodyText'].iloc[0]
55
56 out[19]:
57 "Sue George is the editor of the Guardians International Development Journalism competition, \
58 and was involved in setting it up in 2008. She was an editor at Guardian Creative for 10 years until 2009, \
59 and now works as a freelance journalist and lecturer, specialising in development issues. Her interest in global \
60 development was sparked by the combination of an early job at the BBC World Service, the internationalist feminist \
61 movement of the 1980s, and HIV/Aids activism of the 1990s."
```

Extract documents from NY Times




```
    "docs": [
      {
        "web_url": "https://www.nytimes.com/2017/06/03/world/asia/india-zika-virus.html",
        "snippet": "The government had kept quiet about the cases, hoping to avoid public panic, while stepping up testing an
d mosquito control efforts....",
        "lead_paragraph": "The government had kept quiet about the cases, hoping to avoid public panic, while stepping up tes
ting and mosquito control efforts.",
```

```
  "headline": {
    "main": "India Acknowledges Three Cases of Zika Virus",
    "print_headline": "India Acknowledges Three Cases of Zika Virus"
```



Data selection – Training data

53k documents used as “training data”

Includes some “negative” data

Attempted to balance sample of topics

Topic count still low

Pre-processing *text*

SpaCy

Free open source natural language processing (NLP) library for Python.

```
#installing Spacy
import spacy

#Load spacy
nlp = spacy.load('en_core_web_md')

#using spacy pipeline
en_doc = nlp(u'The French president, François Hollande, has paid a surprise visit to the countrys troops in Afghanistan.')

#print the "lemma", the part of speech tag, stop words
Print([(tok,tok.lemma_, tok.tag_,tok.pos_,tok.is_stop) for tok in en_doc])

Out[42]:
[(The, 'the', 'DT', 'DET', False),
 (French, 'french', 'JJ', 'ADJ', False),
 (president, 'president', 'NN', 'NOUN', False),
 (,, ',', ',', 'PUNCT', False),
 (François, 'françois', 'NNP', 'PROPN', False),
 (Hollande, 'hollande', 'NNP', 'PROPN', False),
 (,, ',', ',', 'PUNCT', False)....
```

NLP

WORDS IN RED REMOVED

The government was yesterday given the go-ahead to appeal against a high court judgment which ruled that the closing of a huge corruption inquiry into a Saudi arms deal was unlawful. The appeal to the House of Lords will delay any possible reopening of the Serious Fraud Office investigation into the allegations for several months.

TOKENISATION, NAMED ENTITIES AND LEMMATIZATION

government	yesterday	give	appeal	high	court	judgement	rule	closing	
huge	corruption	inquiry	saudi	arm	deal	unlawful	house of lords	possible	reopening
the serious fraud office	investigation	allegation	several months						

Transformation: documents to data

```
73 #Transformation of the documents into data
74 vectorizer = TfidfVectorizer(tokenizer=tok, preprocessor=prep,
75                               ngram_range=(1,3), min_df=2)
76
```

EXAMPLE NGRAMS

unigram	government		
bigram	government	yesterday	
trigram	government	yesterday	give

EXAMPLE WORD VECTOR (TF-IDF)

ARTICLE	GOVERNMENT	GOVERNMENT YESTERDAY	GOVERNMENT YESTERDAY GIVE	YESTERDAY	YESTERDAY GIVE	YESTERDAY GIVE APPEAL
X	0.1231	0.1254	0.2371	0.1134	0.1247	0.1247

http://www.tfidf.com/

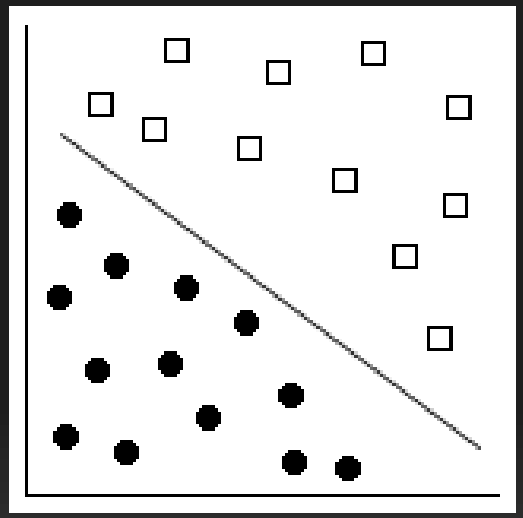
Modelling

Linear support vector classification (sklearn)

One model per topic and hyperplane separating one class against the others

Each example described by set of features

```
#create a new column with topics of interest only
df['Filtered_Topics'] = df['Topics'].apply(lambda topics: [t for t in topics if t in desired_topics])

#sklearn to create Y labels of series of 0 and 1s
from sklearn.preprocessing import MultiLabelBinarizer
mlb = MultiLabelBinarizer(classes=list(topics_df['topics'])) #classes keep all in order of the topic dictionary
Y = mlb.fit_transform(df['Filtered_Topics'])
print(mlb.classes_)

# create a pipeline: pipeline module of scikit-learn allows you to chain transformers and
#estimators together in such a way that you can use them as a single unit.
model = Pipeline([
    ('vectorizer', vectorizer),
    ('classifier',OneVsRestClassifier(LinearSVC(C=100)))
    ])

#create train and test set (test = 1% of training)
docs_train, docs_test, labels_train, labels_test = train_test_split(
            X_train, Y_train, test_size=0.1, random_state=42)

#train model
model.fit(docs_train, labels_train)

#predict model
labels_predict = model.predict(docs_test)
```



Evaluation

Results

Model evaluated using micro F1* score on the test set

Top 20 ranking out of 579 entries

```
90  #print results
91  print(classification_report(labels_test, labels_predict))
92
93  out[20]:
94  Topic                    precision    recall  f1-score   support
95  activism                      0.00      0.00      0.00         0
96  afghanistan                   0.98      0.55      0.70       203
97  aid                           0.92      0.27      0.41        45
98  algerianhostagecrisis         1.00      0.33      0.50         3
99  alqaida                       0.93      0.22      0.36        63
100 alshabaab                     0.00      0.00      0.00         1
```

*http://rushdishams.blogspot.co.uk/2011/08/micro-and-macro-average-of-precision.html

What would I do differently?

Code available here:

https://github.com/k1mmie/datasciencechallenge