Andy Haldane speech analysis

April 23, 2022

1 Andy Haldane speech analysis

The purpose of this task was to use Natural Language processing techniques to analyse and summarise key risks mentioned in the speeches of former Chief Economist of the Bank of England Andy Haldane. The speeches can be found here: https://www.bankofengland.co.uk/news/speeches.

In total, 55 speeches were analysed ranging from 18 August 2011 - 30 June 2021.

1.1 Data import

Fristly, the text was imported from each of the speech PDFs.

Cleansing the text was required to improve the performance of the analysis. The data cleaning performed included:

- removing page numbers
- removing hyperlinks
- removing puncuation (some extra punction was added additional to the default list, like curly speech marks)
- stopwords
- lemmatisation of the text - standardising words, for example banks –> bank

During the analysis, a handful more stopwords were added to the default list to increase the performance of the analysis. The added stopwords were:

- central, bank, et, al, et al, uk, chart, s

At the end of this stage, the corpus consisted of a list of 55 strings of the cleansed text from the speeches.

[1]:
```
from file_import import import_pdf
import glob
from tfidf import *
from co_occurance import *
import numpy as np
import pandas as pd

# load speeches and clean text
corpus=[]
publishing_dates = []
for file in list(glob.glob('speeches/*.pdf')):
```

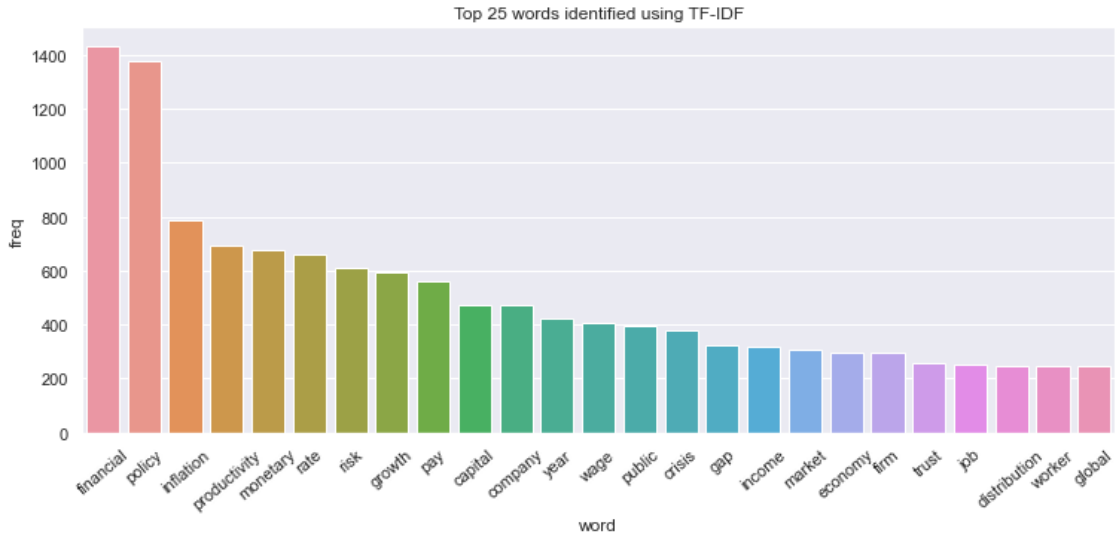
```
    pdf, d = import_pdf(file)
    corpus.append(pdf)
    publishing_dates.append(d)
```

1.2 Term Frequency - Inverse Document Frequency

I chose TF-IDF as a way to get an initial indication of the key risks mentioned in Andy Haldane's speeches. I ran the TF-IDF algorithm across the whole corpus and selected the top 10 results from each speech (550 results total). Then, I counted the frequency of the 10 results in the corresponding speech . Results that appeared in more than 1 speech had their frequencies totaled to get a total frequency for each identified word across all 55 speeches. I selected the top 25 of these to assess whether any risk terms were identified by the algorithm.

[2]:
```
#TF-IDF algorithm
TFIDFvectorizer, feature_names = TFIDF(corpus)
results = corpus_resuts(corpus, TFIDFvectorizer, feature_names, 10)


def corpus_totals(corp, tfidf_results, top_n):
    '''
    Count the frequency of each TF-IDF result in the corresponding speech
    add these and select the top 25 words
    '''
    totals = dict()
    for i in range(len(corp)):

        text=corpus[i].split()

        for j in range(len(tfidf_results[i])):
            tfidf_word = tfidf_results[i][j]
            count = text.count(tfidf_word)

            if tfidf_word in totals:
                totals[tfidf_word] += count

            else:
                totals[tfidf_word] = count

    totals = dict(sorted(totals.items(), key=lambda item: item[1],␣
 ↪reverse=True))
    totals  = {k: totals[k] for k in list(totals)[:top_n]}

    return totals

top_words = corpus_totals(corpus, results, 25)
top_words = pd.DataFrame(top_words.items(), columns=['word','freq'])
```

```
import seaborn as sns
from matplotlib import pyplot as plt
sns.set(rc={'figure.figsize':(10.7,5.27)})
sns.barplot(x='word', y='freq', data=top_words).set_title('Top 25 words␣
 ↪identified using TF-IDF')
plt.xticks(rotation=40)
plt.tight_layout()
plt.show()
```



Based on the results of the TF-IDF, although 15 I deemed to not be risk related, I identified 10 risk words from the top 25 of the TF-IDF keyword extraction:

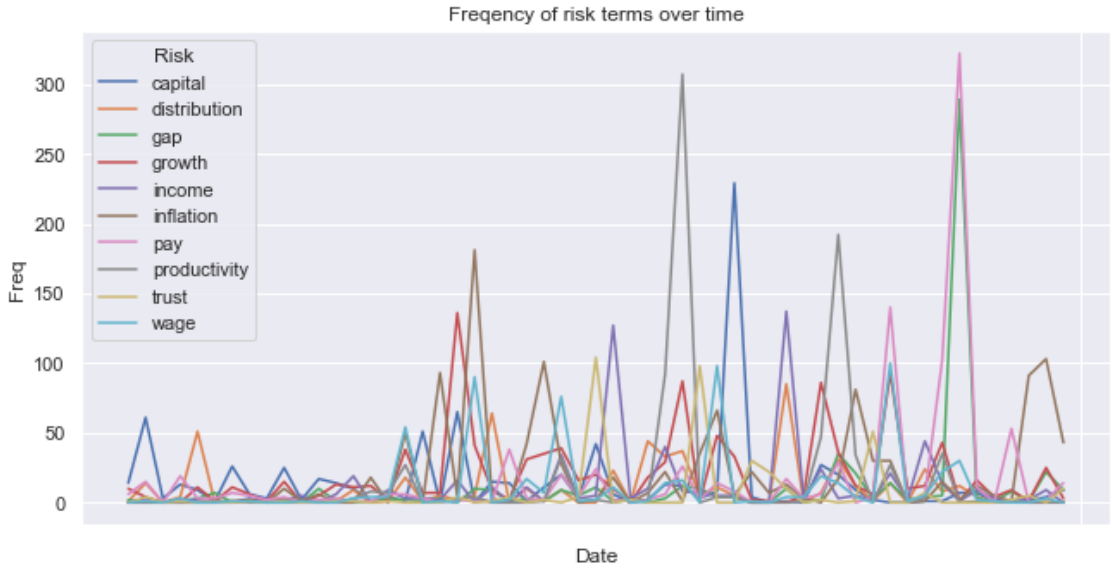
1. inflation
2. productivity
3. growth
4. pay
5. capital
6. wage
7. gap
8. income
9. trust
10. distribution

I expected Brexit or Covid to feature in this list and was surprised they did not appear in the chart above.

1.2.1 Time series plot of risks identified

I next wanted to investigate how these risks might have evolved over time using the dates of publication from the speech PDFs

[3]:
```
def risks_time_series(corp, risk):
    '''
    Count the frequency of each risk term in each document
    join to the date of the speech to create a time series
    for each risk term
    '''
    time_series = pd.DataFrame(columns=['Date', 'Freq', 'Risk'])

    for r in risk:
        r_count = []

        for doc in corp:
            r_count.append(doc.count(r))

        r_ts = pd.DataFrame([publishing_dates, r_count]).transpose()
        r_ts.columns = ['Date', 'Freq']
        r_ts['Risk'] = r

        time_series = time_series.append(r_ts)

    return time_series

risks = ['inflation',
         'productivity',
         'growth',
         'pay',
         'capital',
         'wage',
         'gap',
         'income',
         'trust',
         'distribution']

risk_ts = risks_time_series(corpus, risks)

#wide data for  plotting
risk_ts = risk_ts.pivot("Date", "Risk", "Freq")
sns.set(rc={'figure.figsize':(10.7,5.27)})
sns.lineplot(data=risk_ts, dashes=False).set_title('Freqency of risk terms over␣
 ↪time')
plt.ylabel('Freq')
plt.xticks('')
plt.show()
```



(For readibility, the x labels have been left blank)

There doesn't appear to be a discernible pattern in the the trend of various risk words. A reason for this could be the wide variety of speechs Andy Haldane gives - seminars, annual speeches etc. Restricting the corpus to only speeches to acedemic institutions for example might yield better results.

1.3 Co-Occurrance of words

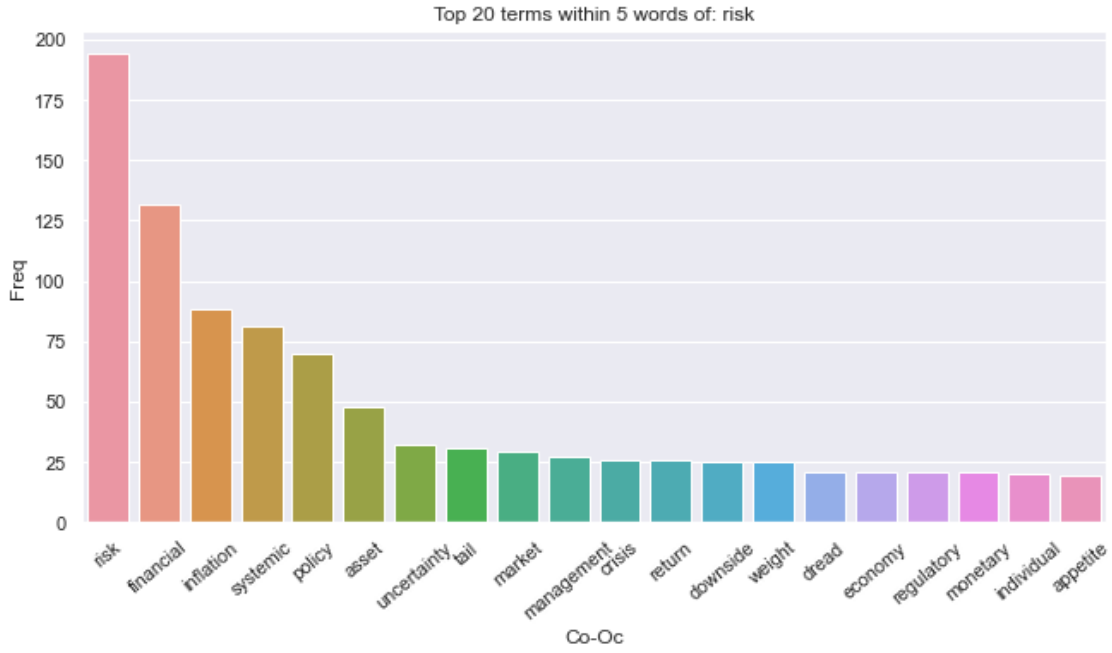
I next wanted to look at the co-occurrance of words with terms that could indicate risk. The terms I decided to assess were:

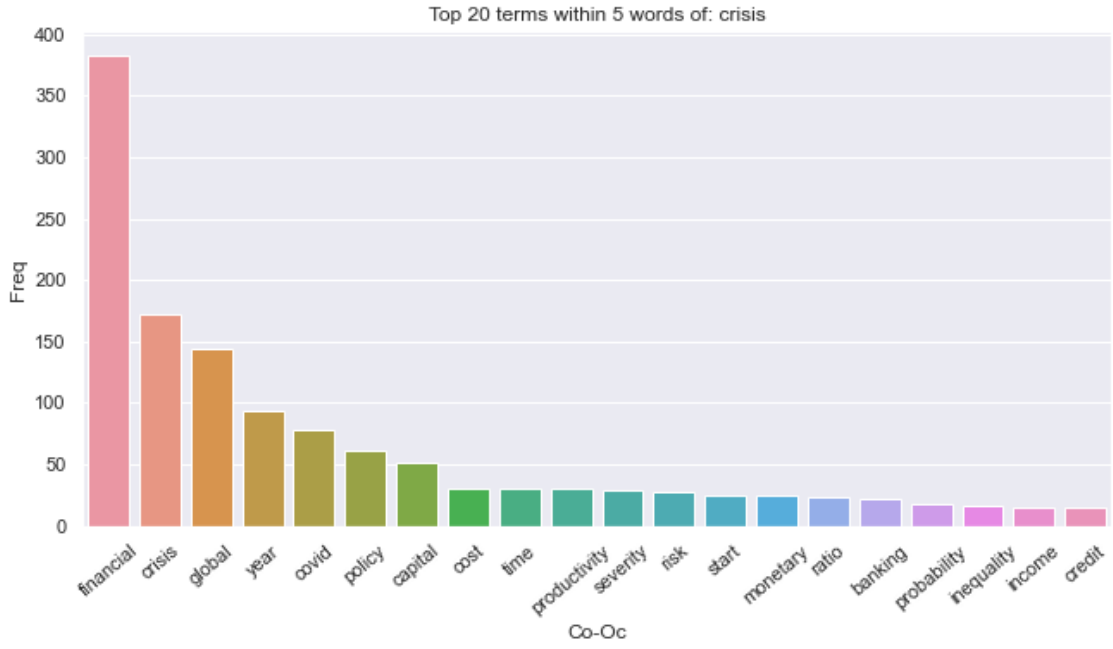
- risk
- crisis
- uncertainty

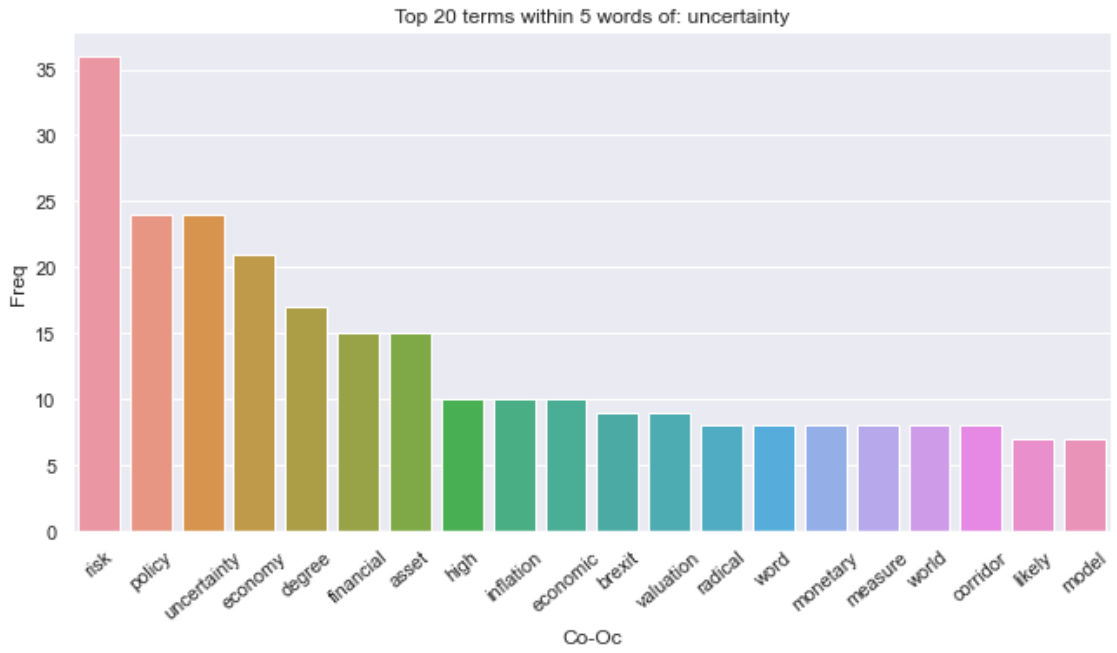
I assessed whether words with a high frequency of occurring within 5 words of a risk term could indicate the subject of risk in the speeches.

[4]:
```
highest_coo = pd.DataFrame()

def top_coocurrance_words(doc, target):
    '''
    Generate a co-occurrance matrix of each speech
    select the top 10 co-occurance terms with the target risk word
    '''

    vocab_dict = build_vocabulary(doc)

    if target in vocab_dict.keys():
```

```
        #generate dataframe of top 10 co-occurance words
        co_ocurrence_vectors = pd.DataFrame(
            np.zeros([len(vocab_dict), len(vocab_dict)]),
            index = vocab_dict.keys(),
            columns = vocab_dict.keys()
        )

        co_ocurrence_vectors = build_context(doc, 5, co_ocurrence_vectors)

        #generate dataframe of top 10 co-occurance words
        top_words = co_ocurrence_vectors.loc[target].
 ↪sort_values(ascending=False).head(10)
        top_words = top_words.rename('Freq')
        top_words = pd.DataFrame(top_words)
        top_words = top_words.rename_axis('Co-Oc').reset_index()
        top_words['Target'] = target

        return top_words

risk_terms = ['risk', 'crisis', 'uncertainty']

for target in risk_terms:
    for doc in corpus:

        top_values = top_coocurrance_words(doc, target)
        highest_coo = highest_coo.append(top_values)

#sum repeated terms
highest_coo = highest_coo.groupby(['Target', 'Co-Oc']).sum().
 ↪sort_values('Freq', ascending=False)
highest_coo = highest_coo.reset_index('Co-Oc')

#select top 20 co-occurrancd words for each target risk term
coo_top20 = highest_coo.groupby('Target').head(20)

for rt in risk_terms:
    title = 'Top 20 terms within 5 words of: ' + rt
    sns.barplot(data=coo_top20.loc[rt], x='Co-Oc', y='Freq').set_title(title)
    plt.xticks(rotation=40)
    plt.show()
```


Top 20 terms within 5 words of: risk
Freq
Co-Oc
risk
financial
inflation
systemic
policy
asset
uncertainty
tail
market
management
crisis
return
downside
weight
dread
economy
regulatory
monetary
individual
appetite


Top 20 terms within 5 words of: crisis
Freq
Co-Oc
financial
crisis
global
year
covid
policy
capital
cost
time
productivity
severity
risk
start
monetary
ratio
banking
probability
inequality
income
credit



There are some intuitive results from the above. 'financial' and 'global' have a high frequency of occurring near 'crisis' and which is probably a result of the phrase 'global financial crisis' occurring across many speeches. Similairy, 'systemic risk' is likely another popualr phrase resulting in a high frequency in the above chart.

'brexit' and 'covid' both appear in one of the above charts, confirming my intuition they should feature somewhere in an analysis of risks contained in speeches. On the other hand, 'covid' would likely only have featured in speeches from 2020, so given the corpus of speeches begins in 2011 and ends mid-2021 'covid' must have a high number of mentions in the speeches from 2020, indicating it is an important recent risk.

2 Conclusion

Comparing the TF-IDF method above and this method, there are some cross-over terms between the two. The matching terms identified are:

- inflation
- productivity

Additionally, due to 'covid' being in the top 5 of word occurrances with 'crisis' despite it likely only becoming featured from 2020, I think this is a key recent risk.

Therefore I believe these are the 3 key risks that feature most prominently across Andy Haldane's speeches.