Reading Between the Lines: Economic Signals and Sentiment from Bank of England (BoE) Speeches

Understanding How BoE Communications Reflect and Influence the UK Economy

Team 14 - Data JAAMN

Jurgita Cepure

Michael Sunner

Nanik Nanwani

Ng Wen Wen Annora

Suleiman Chun Lum Andy Ho

1. Introduction:

The Bank of England (BoE) uses communication by representatives via public speeches as a tool to provide transparency on their monetary policy decisions and economic outlook.

BoE are keen to understand the broader impact of their public speeches. A 5 Why's Framework (*Appendix B*) was used to contextualise and highlight the importance of this issue in order to provide structure to the analysis.

This analysis examines how sentiment of speeches has evolved over time and how sentiment changes before and after key political and economic events. The correlation between sentiment and inflation and GDP will also be explored as well as examining the relationship of sentiment on price movements in 10-year UK Gilts, FX and SONIA markets.

The analysis aims to provide BoE with data-driven insights into the effectiveness and potential consequences of central bank speeches and offer recommendations to refine future communication strategies.

2. Data and Methodology:

2.1 Data Sources

Data Validation and Cleaning - Data validation and cleaning for indicators followed a set of general steps listed in the table below.

Data Validation and Cleaning	
Step	**Procedure**
1	Check and drop missing values
2	Check and drop duplicates (by date column)
3	Drop unnecessary columns
4	Change date column to datetime type
5	Aggregate as required(eg., monthly, quarterly)

2.1.1 BoE Speeches:

Column	**Data**	**Comments**
Reference	Unique reference number for speech	Total of 7721
Country	Country where speech was	Subset selection - filtered for UK only speeches,

	given	sorted by date (initially 1,209 records).
Date	Date of speech given	Converted from object to datetime format
Title	Title of speech	
Author	Person who gave the speech	● Misclassification of some governor speeches in the period of up to mid-2003 was acknowledged by BoE, fixed mislabelled governor authors ● Handling of non-surname 'authors'. Some entries had generic or ambiguous author names (e.g., 'cbi', 'committee'). Where possible, these were reassigned using contextual clues (e.g., speech content, date) (*Appendix A.1*): ○ cbi - 1 speech - Reassigned to King as per reference in this speech ○ governor - 1 speech - Reassigned to King as per date of the speech ○ committee - 1 speech -> left as is ○ industry - 1 speech -> left as is ○ no_info - 32 speeches -> after labeling ones for governors 22 speeches left in author = no_info category; more likely made by other BoF staff hence is_gov=0. ○ summit - 1 speech -> left as is
Is_gov	Boolean Value: 1 = Speech Delivered by governor	Governor attribution fix: ● Reassign author and set is_gov = 1 for governor speeches mistakenly marked otherwise. ● Used date ranges and keyword checks (via change_gov_authorship_by_date() function (*Appendix A.2)*) to map authorship accurately to: ○ George (1993–2003) ○ King (2003–2013) ○ Carney (2013–2020) ○ Bailey (2020–Present)
Text	Transcript of Speech	Duplicate Removal: ● Identified 7 duplicated speeches based on the text field. ● Cross-referenced with the BoE speech

		calendar. ● Dropped those duplicates, reducing records to 1,202 speeches. Lower-cased and removed punctuation from the speech text column using the function clean_text_column(). (*Appendix A.2)*

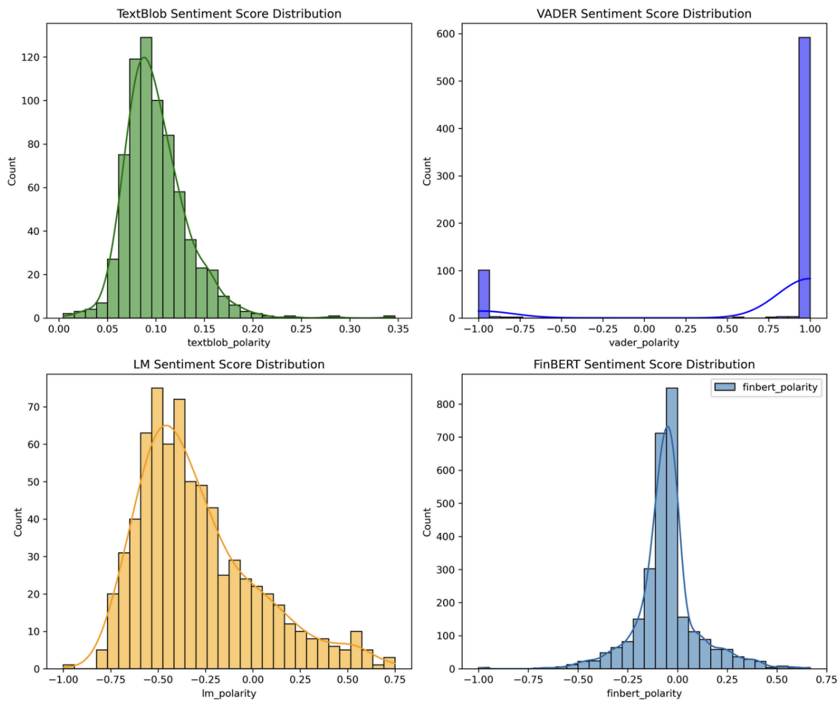
2.1.2 Monetary Policy and Financial Stability Reports:

Column	**Data**	**Comments**
Date	Date of report publication	Converted from object to datetime format
Link	Relative URL links to reports	Main link: https://www.bankofengland.co.uk

Web-scraping Report Release Dates - Release dates of monetary policy reports and financial stability reports were scraped using selenium to extract text from BoE website and BeautifulSoup to parse the text (*Appendix A.4*).

2.2 Sentiment Analysis

NLP Model Selection				
Model	**Training Domain**	**Strengths**	**Limitations**	**Suitability for BoE Speeches**
TextBlob	General English	Easy to use; good for basic sentiment tasks	Too generic; misses financial tone	Low – lacks domain relevance
VADER	Social media text	Captures casual tone; good for polarity detection	Not suited for formal, financial speech	Low – tone mismatch
Loughran-McDonald	Financial lexicons	Tailored for financial texts; transparent scoring	Limited context awareness; rigid vocabulary	Medium – decent fit, but outdated
FinBERT	Financial documents (LLM)	Deep financial context understanding; nuanced sentiment	Requires more resources; less transparent scoring	High – strong contextual and domain fit



Text Preparation for FinBERT - FinBERT model (*Appendix A.5 & A.6*) takes into account contextual information. Text cleaning steps followed those used for other models, with the exception that FinBERT does not require lemmatization or lowercasing. Semi-colons were replaced with full stops to improve sentence tokenization. Speeches were also chunked due to the model's 512-token limit.

2.3 Approach

Broad Sentiment Exploration (Full Dataset) - The initial analysis covered the full archive of Bank of England speeches from 1997 to 2022. Speech frequency was examined over time and by speaker role (Governor vs Staff). Four sentiment analysis models were tested—VADER, TextBlob, Loughran-McDonald, and FinBERT. While most speech components were neutral or cautious in tone, general-purpose models tended to exaggerate sentiment, often misclassifying them as overly positive or negative. As a result, FinBERT was selected for its more domain-appropriate, finance-specific output.

Event-Correlation Testing - To test sentiment responsiveness, speeches were mapped to key internal events—Bank Rate decisions, Monetary Policy Reports, and Financial Stability Reports—as well as major external events like the Global Financial Crisis, Brexit, and COVID-19. Each speech was flagged based on proximity to these events (±7 days for internal, ±1 quarter for external). While no statistically significant sentiment shifts were detected around internal events across any method, some tools did detect some significant - but inconsistent - differences in sentiment around external events. (*Appendix C.1.1)*

Comparison of mean sentiment scores of **ALL** speeches inside and outside **3 internal event** windows

(Bank Rate Decision (±7 days), Financial Stability Report (±7 days), Monetary Policy Report (±7 days))

Event	Tool	Mean Inside	Mean Outside	P-value	Significant?
Bank Rate	VADER	0.732	0.667	0.2343	✕ No
Bank Rate	LM	-0.007	-0.007	0.9751	✕ No
Bank Rate	FinBERT	-0.025	-0.052	0.1105	✕ No
FSR	VADER	0.765	0.727	0.5818	✕ No
FSR	LM	-0.004	-0.007	0.065	✕ No
FSR	FinBERT	-0.035	-0.043	0.7775	✕ No
MPR	VADER	0.737	0.729	0.8943	✕ No
MPR	LM	-0.007	-0.006	0.6829	✕ No
MPR	FinBERT	-0.038	-0.043	0.8241	✕ No

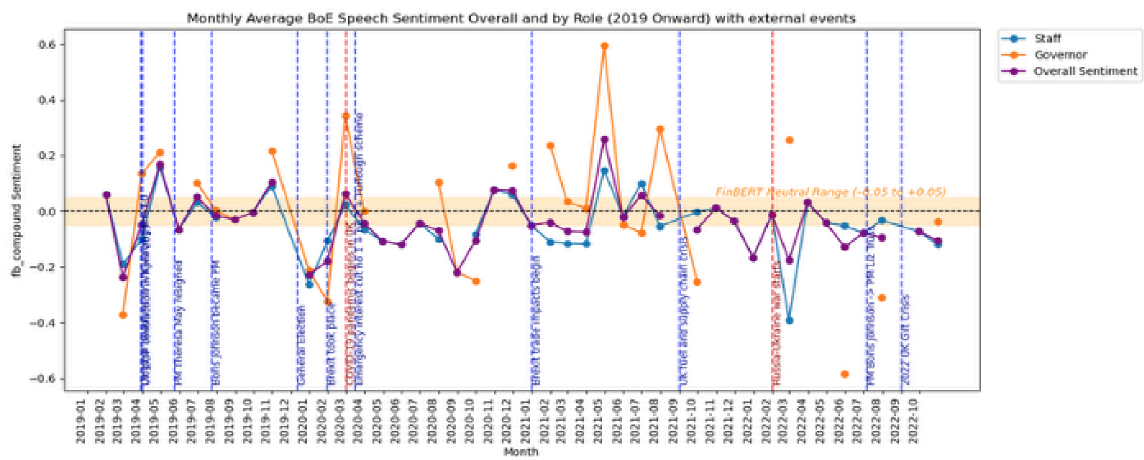
Comparison of mean sentiment scores of **ALL** speeches inside and outside **3 external event** windows

(Global Financial Crisis (GFC): 2007-07-01 → 2015-06-30, Brexit Period: 2016-06-23 → 2020-01-31, COVID-19 Pandemic: 2020-03-01 → 2021-12-31)

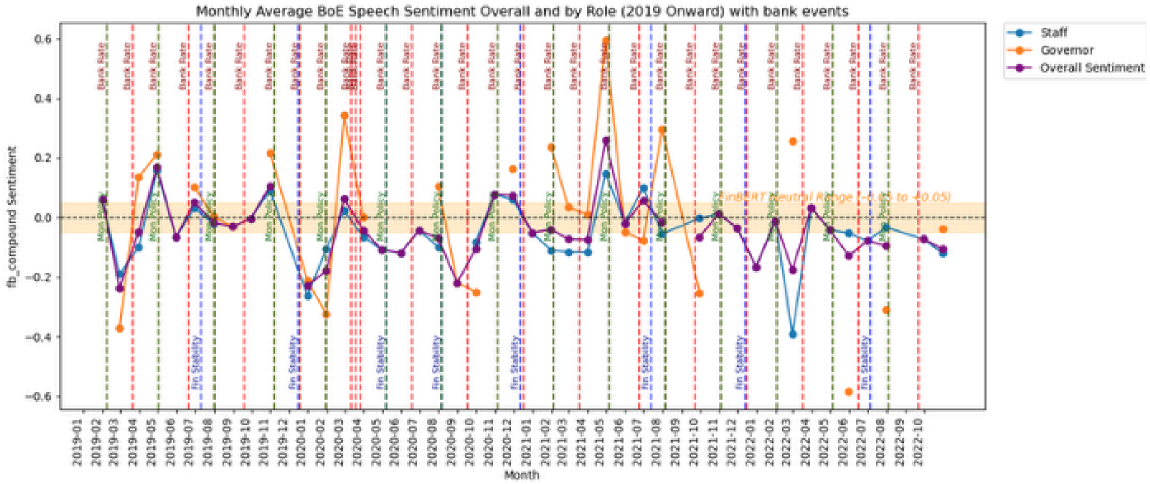
Event	Tool	Mean Inside	Mean Outside	P-value	Significant?
GFC	VADER	0.677	0.696	0.7401	✕ No
GFC	LM	-0.009	-0.005	0.0	✓ Yes
GFC	FinBERT	-0.085	-0.02	0.0001	✓ Yes
Brexit	VADER	0.729	0.664	0.2256	✕ No
Brexit	LM	-0.004	-0.008	0.0	✓ Yes
Brexit	FinBERT	-0.007	-0.065	0.0007	✓ Yes
COVID	VADER	0.622	0.703	0.2781	✕ No
COVID	LM	-0.008	-0.006	0.1607	✕ No
COVID	FinBERT	-0.026	-0.046	0.3423	✕ No

1305_EDA_Speeches10yr_Events_Sentiment.ipynb

<u>Focused Timeframe Analysis</u> - Following a meeting with the Bank of England, the analysis was narrowed to the last 10 years, with a sharper focus on the event-rich period from 2019 to 2022. An updated list of UK-specific events was used, and FinBERT sentiment was examined in greater detail through role-based comparisons (Governor vs Staff) *(Appendix C.1.2)* chi-square tests on sentiment distribution (*Appendix D.1.1*) and analysis of how sentiment varied across speeches tagged with specific topics such as inflation, growth, financial markets, and labour markets.



2105_JC_1it_EDA_2019_22Speeches_Events_Sentiment.ipynb

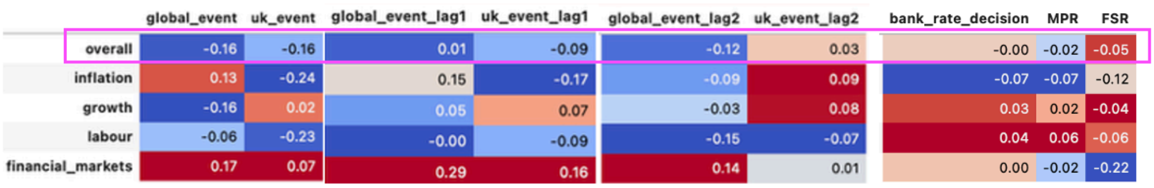


2105_JC_1it_EDA_2019_22Speeches_Events_Sentiment.ipynb

Topic Modelling - Given the lack of clear correlation between overall sentiment and event timing, the approach was refined to focus on topic-specific sentiment. Bespoke keyword-based labelling was introduced to categorize speeches by key economic themes—Growth, Inflation, Financial Markets, and Labour Markets (*Appendix C.3)*. This allowed for a more nuanced and policy-relevant analysis of how sentiment in targeted content aligned with real-world developments.

External and Internal Event Correlation with Sentiment

2105_JC_1it_EDA_2019_22Speeches_Events_Sentiment.ipynb



	global_event	uk_event	global_event_lag1	uk_event_lag1	global_event_lag2	uk_event_lag2	bank_rate_decision	MPR	FSR
overall	-0.16	-0.16	0.01	-0.09	-0.12	0.03	-0.00	-0.02	-0.05
inflation	0.13	-0.24	0.15	-0.17	-0.09	0.09	-0.07	-0.07	-0.12
growth	-0.16	0.02	0.05	0.07	-0.03	0.08	0.03	0.02	-0.04
labour	-0.06	-0.23	-0.00	-0.09	-0.15	-0.07	0.04	0.06	-0.06
financial_markets	0.17	0.07	0.29	0.16	0.14	0.01	0.00	-0.02	-0.22

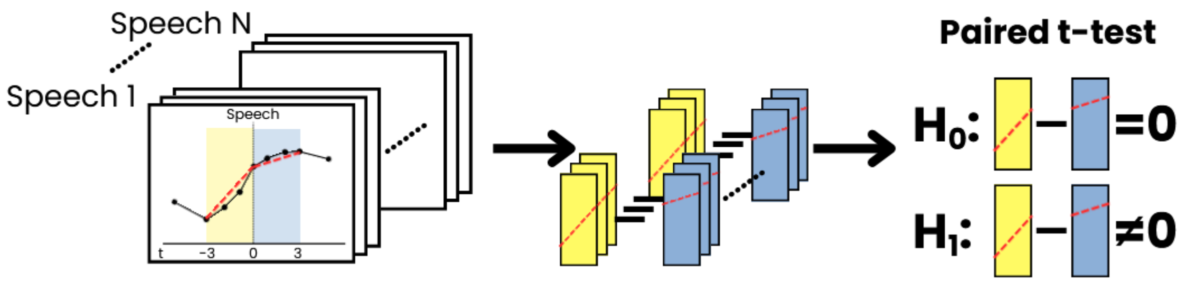
(Code in *Appendix D.1.4*)

Selection of Indicators

Indicator	Reason for Choice	Source
Key Economic Indicators		
Bank Interest Rates	Provided in brief	Bank of England
GDP growth	Specified in Assignment	Office for National Statistics
Inflation (RPI)	Specified in Assignment	Office for National Statistics
Unemployment Rate	Specified in Assignment	Office for National Statistics
Monthly Wages	Specified in Assignment	Office for National Statistics

Labour Productivity	Specified in Assignment	Office for National Statistics
Gilt Yield Curves	Reflects bond market expectations	Bank of England
Market Behavior		
Foreign Exchange (GBP-USD and GBP-EUR)	Captures market reaction to economic signals	Investing.com
Stock Market (FTSE All-Share Index)	Indicator of investor sentiment	Investing.com
Treasury Market (2, 5 and 10 year gilts)	Measures interest rate expectations	UK Debt Management Office (DMO)
Sterling Overnight Index Average	Benchmark for short-term interest rates	Bank of England

Event Related Analysis (ERA) - To explore whether speeches affect economic indicators (especially those with daily frequency), we computed the slope of each indicator in a time window before and after each speech. Paired t-tests were then applied to compare these slopes across all speeches, testing for statistically significant changes in trend.



Cross Correlation - Time series data were cross correlated at different lags to identify the greatest correlation between two time series data and the lag at which it occurred. Data were aligned by date and missing values were forward-filled especially for sentiment scores.

Granger Causality - Granger causality tests whether a time series has predictive information about future values of another. We tested multiple lags and reported the lowest p-values and corresponding lag values. Time series data were aligned by date and missing values were forward-filled. Differencing was applied to ensure stationarity where necessary.

2.4 Refining the Analytical Scope

While the full dataset (1997–2022) showed limited variability in sentiment—even during major internal or external events—we narrowed the analysis to 2012–2022 for higher data consistency and relevance. Within this, we focused specifically on 2019–2022, a period marked by major disruptions including Brexit developments, COVID-19, the Ukraine war, and the gilt crisis.

This window also captured the only statistically significant divergence in sentiment between Governor and Staff speeches (in 2021), with marginal differences in 2019 (*Appendix D.1.1*). This refined window supports the hypothesis that both speaker role and external context influence tone and aims to uncover more meaningful temporal relationships between sentiment shifts and external shocks.

2.5 Analysis Tools and Visualisations:

2.5.1 Analysis Tools

Analysis Tools		
Tool	**Used/Not Used**	**Rationale**
Excel	Yes	Used for initial inspection of the dataset. Helped surface issues like duplicate speech texts and mislabelled authors through manual review. Enabled quick spotting of anomalies before coding.
Python	Yes	Used for data cleaning, sentiment analysis (FinBERT), topic labelling, and statistical tests. Selected for its strong NLP and data science libraries, and suitability for exploratory text analysis.
R	No	Not used. Python provided full coverage for all analysis tasks, including statistical testing and modelling.
Tableau	No	Not required. Visualisations were generated using Python libraries (e.g. matplotlib, seaborn). Interactive dashboards were out of scope.

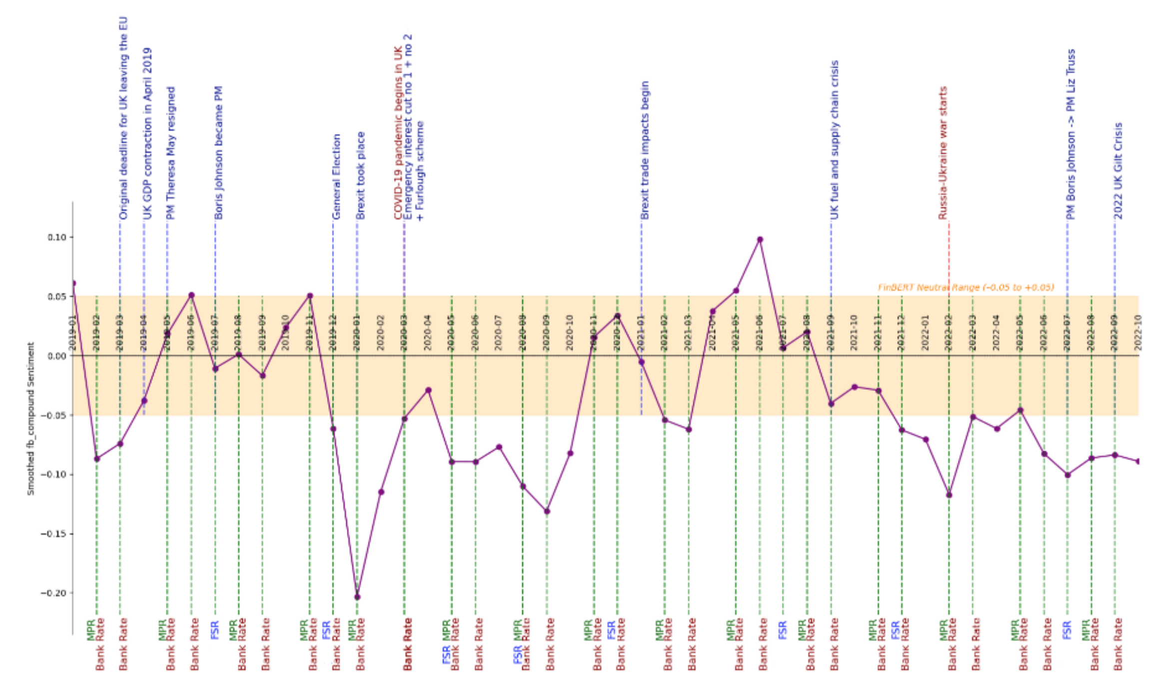
2.5.2 Visualisation Tools and Design Choices

Visualisation Tools and Design Choices	
Tool	**Selection Basis**
Matplotlib/ Seaborn	Used for all core plotting tasks. Provided flexibility to generate time series, box plots, and correlation heatmaps with precise control over format and annotation.
Line Plot (Time Series)	Visualised monthly FinBERT sentiment of Governor vs Staff speeches. Helped highlight role-based divergence and volatility trends over time.
Box Plot (with Paired Lines)	Used to compare GBP-EUR rate changes before vs after staff speeches. Showed significant changes only for neutral sentiment groupings.
Granger Causality Heatmaps	Displayed p-values and lag timings for directional tests between sentiment and macro variables. Helped detect where sentiment might influence or be influenced.
Sentiment Distribution	Showed overall polarity score distribution across all speeches. Used to illustrate BoE's measured, mostly neutral communication tone.

Histogram	
Event Overlay Charts	Visualised major external events (COVID, Brexit, Russia-Ukraine) on time series plots to show when sentiment shifts occurred. Used annotations to provide real-world context.
Correlation Matrix (No Lag)	Visualised pairwise correlation between sentiment and economic indicators (GDP, FX, Gilt Yields). Highlighted low but notable associations.
Lagged Line Charts	Plotted before-and-after sentiment or rate changes (e.g., GBP-EUR) to visualise Granger causality patterns. These charts validated heatmap results by showing actual time series movements aligned with causality lags.
Event Correlation Heatmaps (Lag Based)	Quantified correlation between event types and sentiment across lags. Showed weak statistical influence of events on tone, supporting BoE's consistent messaging strategy.

3. Patterns, Trends & Insights:

3.1 Speech sentiment evolution over time and events:



2105_JC_1it_EDA_2019_22Speeches_Events_Sentiment.ipynb

BoE speeches maintained a neutral tone overall, with only minor shifts even around major events. Even during global crises - like COVID-19 or the Ukraine invasion - BoE speech tone rarely deviated from a narrow neutral band, suggesting a deliberate strategy of acknowledging shocks without emotional drift. Internal events, like rate decisions, also showed little impact.

Across roles, sentiment is most strongly aligned with economic growth - the most common theme - while Governors lean more toward financial market sentiment and Staff toward labour topics, with possible limitation due to sample size in this specific case. Inflation sentiment, however, remains weakly linked, highlighting a consistent emphasis on stability and long-term framing.

Sentiment–Topic Alignment in BoE Communications

Sentiment Role/Topic	Economic Growth	Labour Market	Inflation	Financial Markets	Interpretation
Overall	Strong (≈0.75)	Moderate (≈0.52)	Weak (≈0.12)	Weak (≈0.15)	Growth drives overall tone + moderate labour market sentiment; other topics loosely aligned
Governor	Moderate (≈0.42)	Weak (≈0.13)	Weak (≈0.13)	Moderate (≈0.41)	Governor speeches align with growth + financial market sentiment
Staff	Strong (≈0.69)	Moderate (≈0.47)	Weak (≈0.12)	Weak negative (≈–0.12)	Staff more growth-focused; cautious on markets

2105_JC_1it_EDA_2019_22Speeches_Events_Sentiment.ipynb

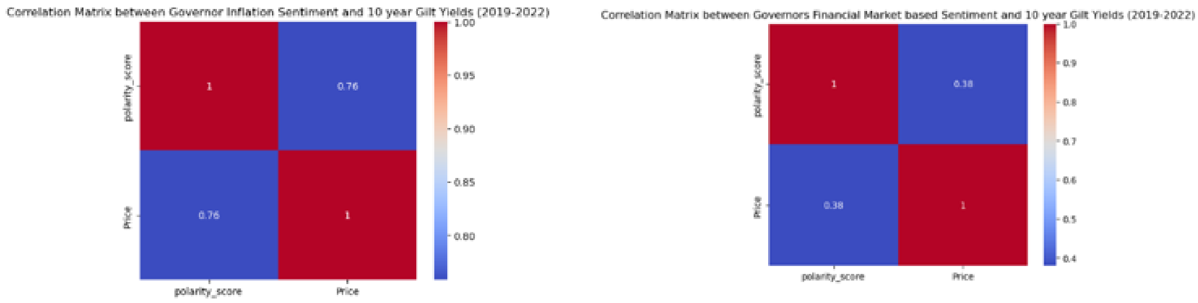
3.2 Sentiment correlations with indicators and topic themes:

Among the indicators referenced in 2.4, the following summarises the most significant insights with additional findings from remaining indicators provided in (*Appendix E).*

<u>3.2.1 10-Year Gilt Market:</u>

When examining if Bank speeches influence the UK 10-year Gilt market, across an expanded time frame of 2012-2022, we found no sustained long-term relationship between the number of speeches, their overall sentiment, and 10-year yield movements.

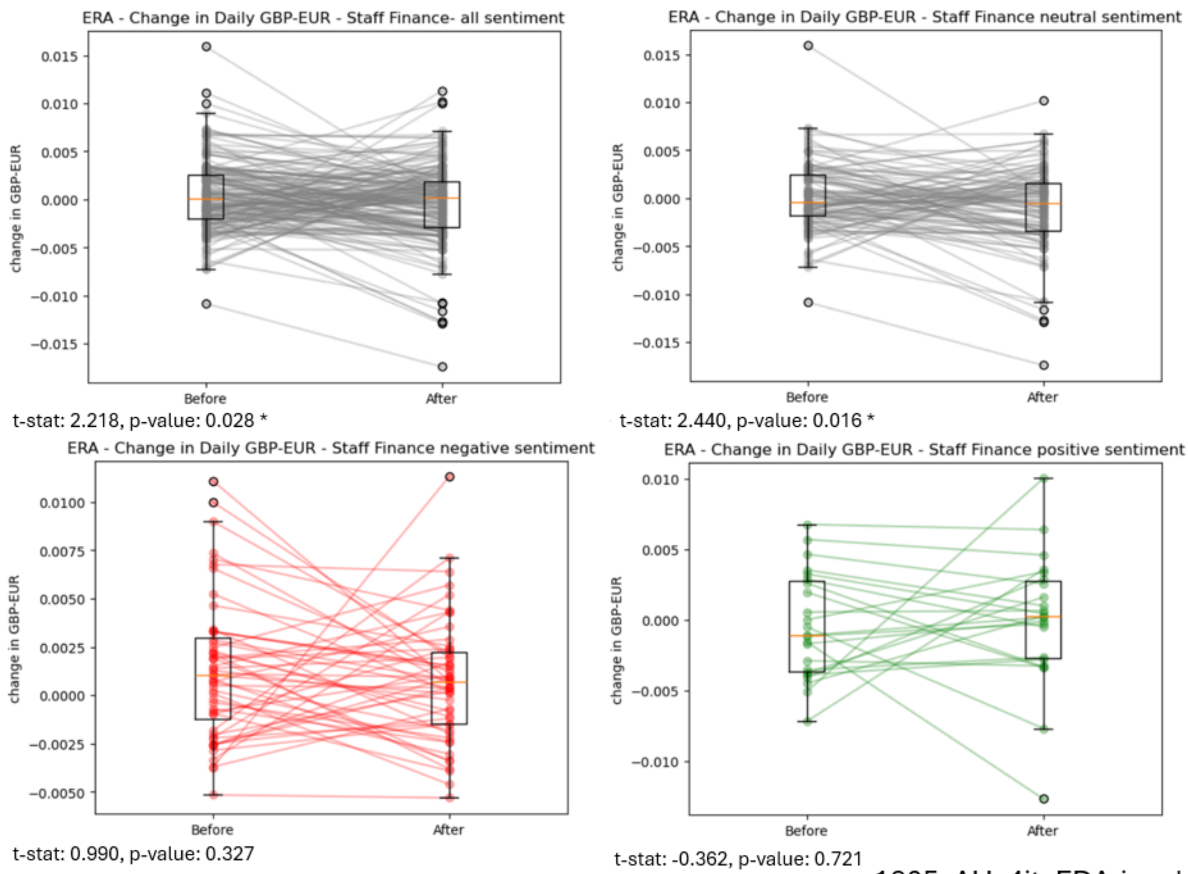
However, during periods of high inflation between 2019 and 2022, we found Governor speeches focused on inflation to have a 76% positive correlation with Gilt yield movements. Speeches focused on financial markets showing a 38% positive correlation, aligning with the overall correlation found between Governor speeches and financial markets. No correlation was observed with staff speeches, despite sharp rises in yields during this timeframe.



18_05_MS_yield_curve_3rd_iteration.ipynb

3.2.2 Foreign Exchange and SONIA

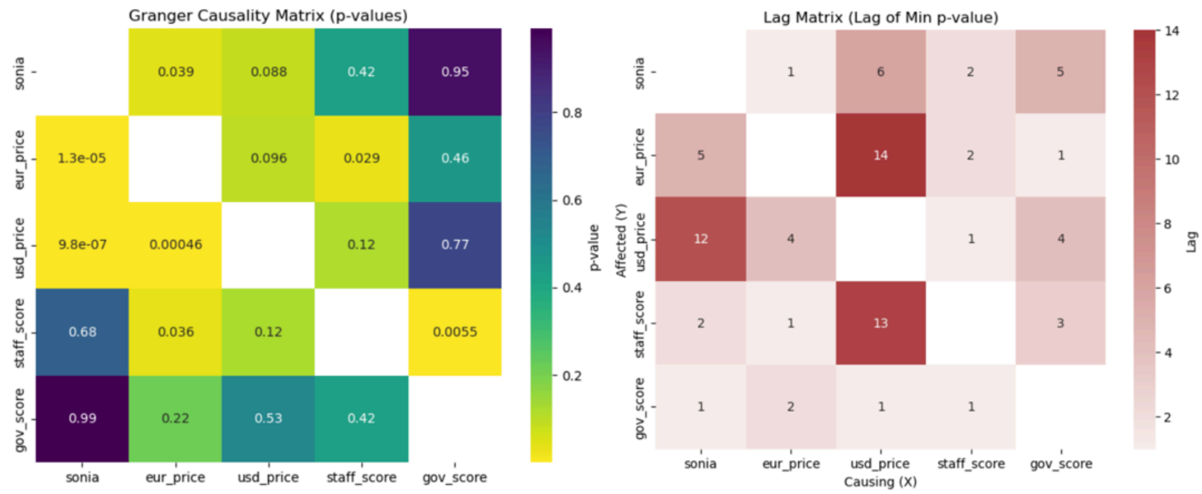
For ERA, statistically significant changes in trajectory were only found for GBP/EUR prices around staff speeches (p=0.028) related to the Financial Market topic (*Appendix C.3*), which were driven by changes surrounding neutral staff speeches (p=0.016). No significant changes were found for GBP/USD and SONIA.



1805_AH_4it_EDA.ipynb

Correlation analysis showed that Financial Market topic FinBERT only had weak correlation with these three indicators (*Appendix D.1.2*).

However, Granger causality tests revealed significant p-values between staff sentiment score and GBP/EUR price meaning staff scores provide significant predictive information about future values of GBP/EUR price and vice versa.

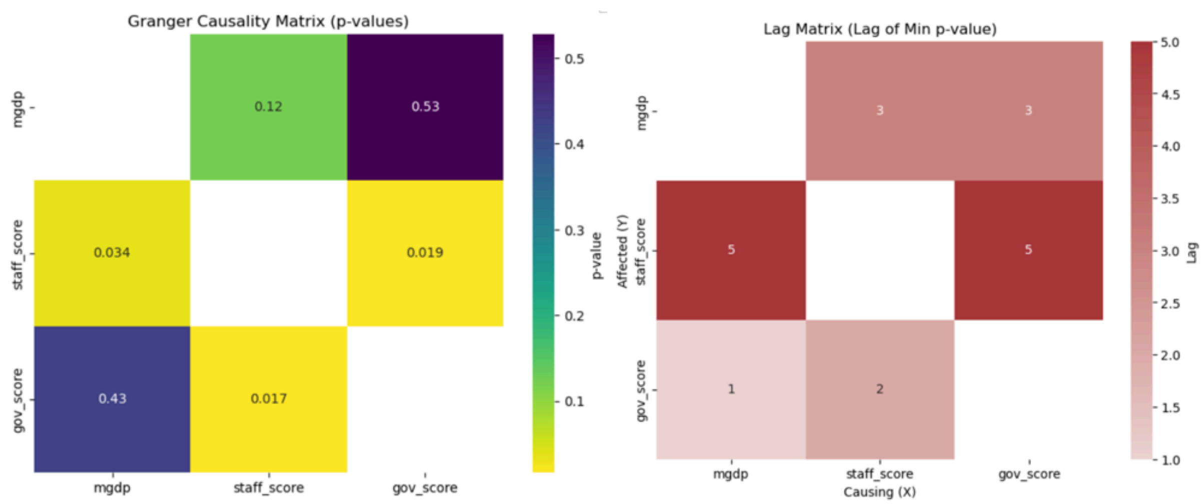


1805_AH_4it_EDA.ipynb

3.2.3 Staff Sentiment and GDP

FinBERT sentiment on the Economic Growth topic showed weak correlation with monthly GDP. (*Appendix D.1.3*).

In contrast, Granger causality indicates that GDP provides significant predictive information about future monthly average staff sentiment scores but this relationship is unidirectional with a lag of five months.

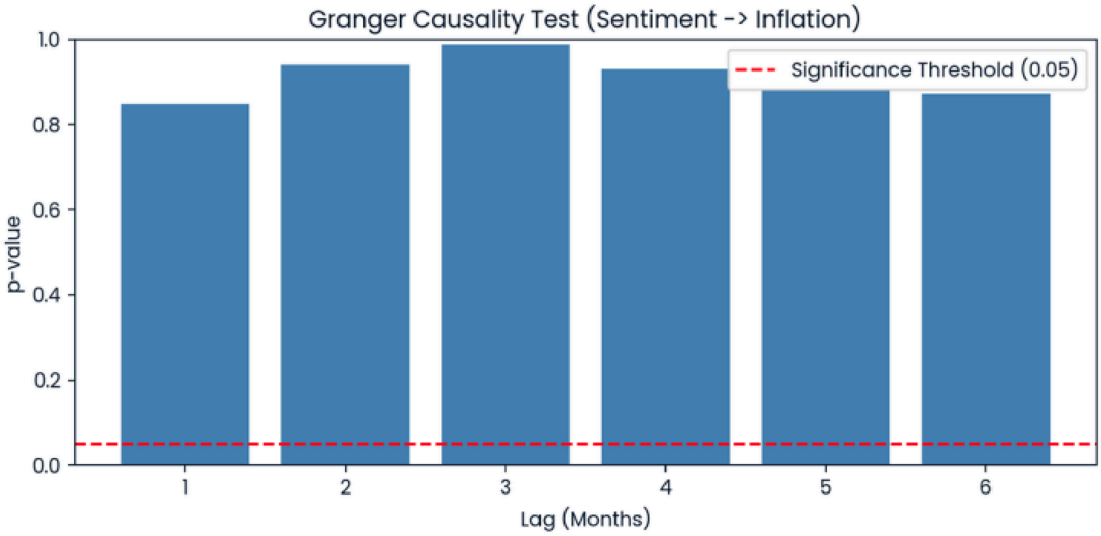


1805_AH_4it_EDA.ipynb

3.2.4 BoE Sentiment and Inflation

To test whether BoE sentiment could predict future inflation, we used a Granger causality test across six monthly lags. The resulting p-values are all well above the 0.05 significance threshold, indicating no statistical evidence that sentiment Granger-causes inflation. This suggests that sentiment is reactive rather than predictive — more likely to reflect current conditions than to signal future price changes.

In short, BoE speech tone may respond to inflationary developments, but it does not appear to anticipate them in a statistically significant way.



4. Conclusions:

In conclusion, our analysis found:

Conclusions:
1. FinBERT to be the most suitable model for capturing financial sentiment, given the distribution was more centred and less extreme than the other sentiment models.
2. Speeches maintained a neutral tone overall throughout the time period, with minor shifts around major economic and political events.
3. External events such as COVID-19 and the Russia–Ukraine war caused brief falls in sentiment, but scheduled BoE actions had limited influence on tone.
4. Tone divergence existed between governor and staff speeches, with governor sentiment being more volatile, expressive and showing strong relationships with indicators such as 10-year Gilts and GBP-EUR FX rate.
5. For other indicators, economic growth sentiment consistently aligned with speech tone, while inflation sentiment remained weakly connected.

5. Recommendations:

We propose the following actions, prioritised by impact and feasibility:

Recommendations	Priority
Integrate sentiment tracking into internal dashboards Incorporate FinBERT sentiment scores to monitor the tone of Bank speeches over time. Significant shifts or sustained deviations from neutral can act as early indicators of changing sentiment, supporting more timely internal awareness and response.	**High**
Contextualise sentiment by speaker role and topic Build reference profiles for Governor and Staff speeches, based on typical tone and themes across key topics. Future speeches can then be compared against these profiles to identify anomalies or shifts in communication strategy. This approach can also help assess consistency during periods of economic or political uncertainty.	**Medium**
Apply sentiment insights to scenario planning Leverage historical sentiment patterns to inform communication strategies under different macroeconomic scenarios. This can help guide tone and messaging during future shocks, supporting coherent and calibrated public communication.	**Low**

6. Limitations & Next Steps:

As a next step, FinBERT sentiment analysis can be extended to Financial Stability and Monetary Policy reports to assess how their tone correlates with speech sentiment.

We recommend advancing topic modelling using Latent Dirichlet Allocation (LDA) to uncover additional themes in Bank communications. This would also help refine the current FinBERT topic labelling approach, which was limited by a manually defined keyword list.

Our analysis focused on speeches from 2019–2022. Future work can expand this to earlier periods—such as before and after the 2008 financial crisis—to test whether current findings hold over time. Additional correlations can also be explored with indicators like the UK's trade balance with the US, EU, and China.

One key limitation of this study is the broad grouping of "Staff" speeches. Tone differences may exist within this category based on seniority or function. A more granular breakdown could reveal important variations not captured in the current analysis.

APPENDIX:

A. Data Cleaning Steps

A.1 Handling of Non-Surname ‘authors’

Example Script for Handling of Non-surname 'authors'

0205_JC_BankSpeeches_extra14_Clean.ipynb

3.4.2. author = governor (1 speech)

```
# Check the speech allocated to author = 'governor'
print(df_uk[df_uk['author'] == 'governor'][['date', 'title', 'text']])
```

```
        date   title                                            text
6 1998-12-01  Speech  It is both an honour and a challenge to delive...
```

```
# The speech date 1998-12-01.
# 'author' reffered to as governor

# Reassign authorship to 'george'by running the function change_gov_authorship_by_date.
df_uk = change_gov_authorship_by_date(
    df_uk,
    keyword='governor',
    text_column='author',
    base_author= 'governor'
)
# Check the number of governors + speeches assigned to each.
df_uk[df_uk['is_gov'] == 1] ['author'].value_counts()
```

```
author
carney    69
king      65
george    56
bailey    55
Name: count, dtype: int64
```

3.4.3.ii author = no_info & title contains 'Mark Carney'

found 5 speeches -> reassigned authorship to 'carney' as per date

```
# Find all speeches where there's word 'Mark Carney'in title for review.
governor_check = (
    (df_uk['is_gov'] == 0) &
    (df_uk['author'] =='no_info') &
    (df_uk['title'].str.contains("Mark Carney", case=False))
)
print(df_uk.loc[governor_check,['date', 'title']])
```

```
          date                                              title
587 2013-08-28  Mark Carney: Speech and Press Conference, held...
602 2013-10-24  Mark Carney: Speech as part of the Financial T...
641 2014-06-12  Mark Carney's Speech at the Mansion House Bank...
653 2014-09-09  Mark Carney's speech at the Trades Union Congr...
918 2018-04-13  Slides from Mark Carney's speech at the Public...
```

```
#5 x speeches date between 2013-08-28 and 2018-04-13.
#'title' has reference to Mark Carney.

#Reassign authorship to 'carney' by runing the function change_gov_authorship_by_date.
df_uk = change_gov_authorship_by_date(
    df_uk,
    keyword= 'Mark Carney',
    text_column='title',
    base_author='no_info'
)

# Check the number of governors + speaches assigned to each.
df_uk[df_uk['is_gov'] == 1]['author'].value_counts()
```

```
author
carney    74
king      69
george    56
bailey    55
Name: count, dtype: int64
```

A.2 Function change_gov_authorship_by_date

0205_JC_BankSpeeches_extra14_Clean.ipynb

```
def change_gov_authorship_by_date(
    df,
    keyword,
    text_column,
    base_author
):
    """
    This function changes the author to the respective governor and updates is_gov,
    based on specific parameters, if it finds the speech was incorrectly labelled
    as not given by a governor.
    """
    role_check = (
        (df['is_gov'] == 0) &
        (df['author'] == base_author) &
        (df[text_column].str.contains(keyword, case=False, na=False))
    )

    df.loc[role_check & (df['date'] > '1993-07-01') & (df['date'] <= '2003-06-30'), ['author', 'is_gov']] = ['george', 1]
    df.loc[role_check & (df['date'] >= '2003-07-01') & (df['date'] <= '2013-06-30'), ['author', 'is_gov']] = ['king', 1]
    df.loc[role_check & (df['date'] >= '2013-07-01') & (df['date'] <= '2020-03-15'), ['author', 'is_gov']] = ['carney', 1]
    df.loc[role_check & (df['date'] >= '2020-03-16'), ['author', 'is_gov']] = ['bailey', 1]

    return df
```

A.3 Function clean_text_column

0205_JC_BankSpeeches_extra14_Clean.ipynb

```
def clean_text_column(df, column):
    """
    Converts text to lowercase and removes punctuation in the specified column.
    """
    df[column] = df[column].apply(lambda x: " ".join(str(x).lower().split()))
    df[column] = df[column].str.replace(r'[^\w\s]', '', regex=True)
    return df
```

A.4 Web Scraping Report Dates

```
# import the required library
from selenium import webdriver

# initialize an instance of the chrome driver (browser)
driver = webdriver.Chrome()

# Target Website
#boe_url="https://www.bankofengland.co.uk/financial-stability-report/financial-stability-reports" #financial stability reports
boe_url="https://www.bankofengland.co.uk/monetary-policy-report/monetary-policy-report" #Monetary Policy (from 2019 onwards)
#boe_url="https://www.bankofengland.co.uk/inflation-report/inflation-reports" #Inflation Report (name of monetary policy before 2019)

driver.get(boe_url)

# output the full-page HTML
print(driver.page_source)
```

```
<html lang="en" class="menu-closed flexboxlegacy flexwrap flexbox objectfit object-fit"><!--<![endif]--><head>
        <!-- 89rht -->

        <link rel="preload" as="style" href="/styles/boe.vendor.min.css?ver=5287d">
<link rel="preload" as="style" href="/styles/BoE.min.css?ver=5287d">
<link rel="preload" as="style" href="/-/media/css/cludosearchmin.css?ver=212A61">


        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta http-equiv="Content-type" content="text/html; charset=utf-8">
        <meta content="width=device-width, initial-scale=1.0" name="viewport">

<title>Financial Stability Reports | Bank of England</title>
<meta name="description" content="">
<meta name="apple-mobile-web-app-title" content="Bank of England">
<meta name="format-detection" content="telephone=no">
<link rel="canonical" href="https://www.bankofengland.co.uk/financial-stability-report/financial-stability-reports">
<meta property="og:title" content="Financial Stability Reports">
```

```
# Now pipe it into BS4
# Accept/Reject cookies first to allow page to load
soup1 = BeautifulSoup(driver.page_source, 'html.parser')
```

```
# Visualize if necessary
print(soup1.prettify())
```

```
<html class="menu-closed flexboxlegacy flexwrap flexbox objectfit object-fit" lang="en">
 <!--<![endif]-->
 <head>
  <!-- 89rht -->
  <link as="style" href="/styles/boe.vendor.min.css?ver=5287d" rel="preload"/>
  <link as="style" href="/styles/BoE.min.css?ver=5287d" rel="preload"/>
  <link as="style" href="/-/media/css/cludosearchmin.css?ver=212A61" rel="preload"/>
  <meta content="IE=edge" http-equiv="X-UA-Compatible"/>
  <meta content="text/html; charset=utf-8" http-equiv="Content-type"/>
  <meta content="width=device-width, initial-scale=1.0" name="viewport"/>
  <title>
   Financial Stability Reports | Bank of England
  </title>
  <meta content="" name="description"/>
  <meta content="Bank of England" name="apple-mobile-web-app-title"/>
  <meta content="telephone=no" name="format-detection"/>
  <link href="https://www.bankofengland.co.uk/financial-stability-report/financial-stability-reports" rel="canonical"/>
  <meta content="Financial Stability Reports" property="og:title"/>
```

0405_Finanical_Stability_Monetary_Policy_Dates.ipynb

A.5 Functions for Topic Modeled FinBERT Sentiment Analysis

```python
def extract_target_sentences(sentences, target_words=target_words):
    """
    Extracts target sentences based on user defined list of target words
    Accepts a list of sentences which are the output of sentence tokenization
    Returns the index of the sentence and the target sentence

    """
    target_pattern = r'\b(?:' + '|'.join(map(re.escape, target_words)) + r')\b'
    # Create a tuple (indices, target sentence)
    target_sentences = [(i, sent) for i, sent in enumerate(sentences) if re.search(target_pattern, sent, flags=re.IGNORECASE)]

    return target_sentences
```

```python
# Function to create chunk text around target sentences
def chunk_text(text, win):
    """
    Separates text of interest into smaller chunks since finBert model can only accept 512 tokens at a time
    Extracts a chunk of text that encapsulates 'win' number of sentences before and after target sentence.
    Sequentially shortens chunk of text while chunk has more than 512 tokens

    """
    token_length=[]
    chunks=[]
    start_index=0

    sentences = sent_tokenize(text)
    max_index=len(sentences)-1
    target_sentences= extract_target_sentences(sentences, target_words)

    for i in range(0,len(target_sentences)):
        start_index = max(target_sentences[i][0] - win, 0)
        end_index = min(target_sentences[i][0] + win + 1, max_index + 1)

        chunk_text=" ".join(sentences[start_index:end_index])
        while len(tokenizer.encode(chunk_text, add_special_tokens=True))>=512:
            end_index-=1

            # Join sentences with a space in between sentences
            chunk_text=" ".join(sentences[start_index:end_index])
        chunks.append(chunk_text)

    return chunks
```

```python
# Classification function for long text
def classify_text(text, pipe, tokenizer, win=2):
    """
    Performs sentiment analysis using finBert model using Pipe
    """
    #Extract label names
    labels = ['neutral', 'positive', 'negative']
    chunks = chunk_text(text, win)
    scores = []

    for chunk in chunks:
      #  print(f'Number of Tokens: {len(chunk)}')
        result = pipe(chunk)[0]
        #print(result) #to keep track of results
        # Map label to score
        label_score_dict = {r['label'].lower(): r['score'] for r in result}

        # Ensure all labels are included and aligned
        aligned_scores = [label_score_dict.get(label, 0.0) for label in labels]
        scores.append(aligned_scores)

    # Average scores per label
    avg_scores = np.mean(scores, axis=0)

    # Convert to Series for easier polarity calculation
    sentiment_scores = pd.Series(avg_scores, index=labels)
    polarity_score = sentiment_scores['positive'] - sentiment_scores['negative']
    sentiment_scores['polarity_score'] = polarity_score

    print(polarity_score)
    return sentiment_scores
```

1305_AH_iter4_Topic_FinBert_SentimentAnalysis.ipynb

A.6 Example Use of Topic Modeled FinBERT Sentiment Analysis

Define Target Words

```
# Define Target Words for Topic
target_words= ['GDP', 'FTSE', 'output', 'productivity', 'investment', 'unemployed', 'recession',
               'economic', 'recovery', 'growth', 'demand', 'supply', 'expansion', 'consumption',
               'slowdown', 'headwinds', 'spending', 'consumer confidence']
```

Load pretrained finBert Model

```
#yiyanghkust/finBert-tone model
finbert = AutoModelForSequenceClassification.from_pretrained('yiyanghkust/finbert-tone')
tokenizer = AutoTokenizer.from_pretrained('yiyanghkust/finbert-tone')

nlp = pipeline("text-classification", model=finbert, tokenizer=tokenizer, top_k=3)
```

```
Device set to use cpu
```

Run finBert Analysis

```
# Run finBert for all speeches

counter=0
length=len(df_uk.text)
sentiment_results=pd.DataFrame([])
# Loop over all speeches
for speech in df_uk.text:
    counter+=1
    print(f'Progress: {counter}/{length}')
    results=classify_text(speech, nlp, tokenizer, win=2)
    sentiment_results=pd.concat([sentiment_results, pd.DataFrame(results).transpose()])
```

```
Progress: 1/714
-0.28285933503341887
Progress: 2/714
-0.45152662819455147
Progress: 3/714
-0.6744195555058727
Progress: 4/714
-0.7690354464797565
Progress: 5/714
nan
Progress: 6/714
C:\Users\Andy\anaconda3\Lib\site-packages\numpy\core\fromnumeric.py:3504: RuntimeWarning: Mean of empty slice.
  return _methods._mean(a, axis=axis, dtype=dtype,
C:\Users\Andy\anaconda3\Lib\site-packages\numpy\core\_methods.py:129: RuntimeWarning: invalid value encountered in scalar divide
  ret = ret.dtype.type(ret / rcount)
0.4509380898799463
Progress: 7/714
```

1305_AH_iter4_Topic_FinBert_SentimentAnalysis.ipynb

B. 5 Whys Framework

Central bank speeches

Why do Central Banks make public speeches?

Speeches discuss the overall state of the economy and add depth and clarity to their monetary policy decisions. Central Banks effectively use speeches as an instrument of monetary policy.

Why do they discuss this?

They do so to provide transparent information to financial markets and the public on economic outlook and monetary policy decisions.

Why is this important?

This helps promote stability in markets (prevent market turmoil) and reassures the public by explaining their activities and economic views.

Why is this important?

Their insights can influence market expectations of future asset prices and provide further depth into their reasoning for their economic views to strengthen public confidence and reduce economic uncertainty. (SUERF Monetary Policy Publication)

Why is this the case?

Speeches are closely followed by market participants and policymakers and every word is scrutinised. Speeches are analysed by not only what has been said but also has not been said. They provide forward guidance and given policy-makers and market participants act in a forward looking manner, speeches can influence future expectations of interest rates/inflation can impact the economy. (Jenss Weidmann & ScienceDirect Article)

C. NLP Model Selection

C.1 Model Performance

C.1.1 Example T-Test Calculations for Sentiment Score Comparison Table

Example T-Test Calculations for Sentiment Score Comparison Table

1305_EDA_Speeches10yr_Events_Sentiment.ipynb

Bank rate decision all speeches FinBERT

```
#### Filter FinBERT compound sentiment based on is_bank_rate_window
inside = df_uk[df_uk['is_bank_rate_window'] == 1]['fb_compound']
outside = df_uk[df_uk['is_bank_rate_window'] == 0]['fb_compound']

# Calculate means
mean_inside = inside.mean()
mean_outside = outside.mean()

# Run t-test
t_stat, p_val = ttest_ind(inside, outside, equal_var=False, nan_policy='omit')

# Print results
print("FinBERT Compound Sentiment: Bank Rate Decision Window")
print(f"Mean Inside Window:  {mean_inside:.3f}")
print(f"Mean Outside Window: {mean_outside:.3f}")
print(f"T-statistic: {t_stat:.3f}, P-value: {p_val:.4f}")

if p_val < 0.05:
    print("→ Statistically significant ✓")
else:
    print("→ Not statistically significant ✗")
```

```
FinBERT Compound Sentiment: Bank Rate Decision Window
Mean Inside Window:  -0.025
Mean Outside Window: -0.052
T-statistic: 1.599, P-value: 0.1105
→ Not statistically significant ✗
```

Financial Stability Reports publication all speeches LM

```
# All speeches
inside = df_uk[df_uk['is_fsr_window'] == 1]['lm_net_sentiment']
outside = df_uk[df_uk['is_fsr_window'] == 0]['lm_net_sentiment']

mean_inside = inside.mean()
mean_outside = outside.mean()
t_stat, p_val = ttest_ind(inside, outside, equal_var=False, nan_policy='omit')

print("LM Net Sentiment: Financial Stability Report Window")
print(f"Mean Inside Window:  {mean_inside:.3f}")
print(f"Mean Outside Window: {mean_outside:.3f}")
print(f"T-statistic: {t_stat:.3f}, P-value: {p_val:.4f}")
print("→ Statistically significant ✓" if p_val < 0.05 else "→ Not statistically significant ✗")
```

```
LM Net Sentiment: Financial Stability Report Window
Mean Inside Window:  -0.004
Mean Outside Window: -0.007
T-statistic: 1.877, P-value: 0.0650
→ Not statistically significant ✗
```

Monetary Policy Reports publication VADER

```
# VADER compound sentiment inside vs. outside the Monetary Policy Reports (MPR) window, including the t-test.

# Filter speeches inside and outside the MPR window
inside = df_uk[df_uk['is_mpr_window'] == 1]['vader_compound']
outside = df_uk[df_uk['is_mpr_window'] == 0]['vader_compound']

# Calculate means
mean_inside = inside.mean()
mean_outside = outside.mean()

# Run t-test
t_stat, p_val = ttest_ind(inside, outside, equal_var=False, nan_policy='omit')

# Display results
print("VADER Compound Sentiment: Monetary Policy Report (MPR) Window")
print(f"Mean Inside Window:  {mean_inside:.3f}")
print(f"Mean Outside Window: {mean_outside:.3f}")
print(f"T-statistic: {t_stat:.3f}, P-value: {p_val:.4f}")

if p_val < 0.05:
    print("→ Statistically significant ✓")
else:
    print("→ Not statistically significant ✗")
```

```
VADER Compound Sentiment: Monetary Policy Report (MPR) Window
Mean Inside Window:  0.716
Mean Outside Window: 0.685
T-statistic: 0.373, P-value: 0.7098
→ Not statistically significant ✗
```

C.1.2 Monthly Average BoE Speech Sentiment Overall and by Role (2019 Onward) with external and bank events

Monthly Average BoE Speech Sentiment Overall and by Role (2019 Onward) with external and bank events

2105_JC_1it_EDA_2019_22Speeches_Events_Sentiment.ipynb

```python
# Define date range
start_date = '2019-01-01'
end_date = df_uk['date'].max()

# Group by month and is_gov role, then calculate average sentiment
df_sentiment_by_role = (
    df_uk[df_uk['date'] >= start_date]
    .groupby([pd.Grouper(key='date', freq='ME'), 'is_gov'])['fb_compound']
    .mean()
    .unstack()  # Create separate columns for each role
)

# Rename columns for legend
df_sentiment_by_role.columns = df_sentiment_by_role.columns.map({1: 'Governor', 0: 'Staff'})

# Filter events
df_ev_global_filtered = df_ev_global[
    (df_ev_global['Estimated Start Date'] >= start_date) &
    (df_ev_global['Estimated Start Date'] <= end_date)
]

df_ev_uk_filtered = df_ev_uk[
    (df_ev_uk['Estimated Start Date'] >= start_date) &
    (df_ev_uk['Estimated Start Date'] <= end_date)
]

# Plot
fig, ax = plt.subplots(figsize=(14, 6))
df_sentiment_by_role.plot(ax=ax, marker='o')
df_sentiment_monthly.plot(ax=ax, marker='o', color='purple', label='Overall Sentiment') #purple

# Add FinBERT neutral band shading (-0.05 to +0.05)
ax.axhspan(-0.05, 0.05, color='orange', alpha=0.2)
ax.text(
    df_sentiment_by_role.index[-12], 0.055,
    "FinBERT Neutral Range (-0.05 to +0.05)",
    fontsize=10, color='darkorange', style='italic'
)

# Add vertical lines for global events
for _, row in df_ev_global_filtered.iterrows():
    ax.axvline(row['Estimated Start Date'], color='red', linestyle='--', alpha=0.7)
    ax.text(row['Estimated Start Date'], df_sentiment_by_role.max().max() * -1.05, row['Event'],
            rotation=90, fontsize=9, color='darkred', va='bottom')

# Add vertical lines for UK events
for _, row in df_ev_uk_filtered.iterrows():
    ax.axvline(row['Estimated Start Date'], color='blue', linestyle='--', alpha=0.7)
    ax.text(row['Estimated Start Date'], df_sentiment_by_role.max().max() * -1.05, row['Event'],
            rotation=90, fontsize=9, color='darkblue', va='bottom')

# X-axis setup
months = pd.date_range(start=start_date, end=end_date, freq='1MS')
ax.set_xticks(months)
ax.set_xticklabels([dt.strftime('%Y-%m') for dt in months], rotation=90)

# Final layout
ax.set_title("Monthly Average BoE Speech Sentiment Overall and by Role (2019 Onward) with external events")
ax.set_xlabel("Month")
ax.set_ylabel("fb_compound Sentiment")
ax.axhline(0, color='black', linestyle='--', linewidth=1)
plt.legend(loc='upper left', bbox_to_anchor=(1.02, 1), borderaxespad=0)
ax.grid(False)
plt.tight_layout()
plt.show()
```

```python
#### Define date range
start_date = '2019-01-01'
end_date = df_uk['date'].max()

# Group by month and is_gov role, then calculate average sentiment
df_sentiment_by_role = (
    df_uk[df_uk['date'] >= start_date]
    .groupby([pd.Grouper(key='date', freq='ME'), 'is_gov'])['fb_compound']
    .mean()
    .unstack()  # Separate columns for Governor and Staff
)

# Rename columns for legend
df_sentiment_by_role.columns = df_sentiment_by_role.columns.map({1: 'Governor', 0: 'Staff'})

# Compute overall monthly sentiment for context
df_sentiment_monthly = (
    df_uk[df_uk['date'] >= start_date]
    .groupby(pd.Grouper(key='date', freq='ME'))['fb_compound']
    .mean()
)

# Plot
fig, ax = plt.subplots(figsize=(14, 6))
df_sentiment_by_role.plot(ax=ax, marker='o')
df_sentiment_monthly.plot(ax=ax, marker='o', color='purple', label='Overall Sentiment')

# Add FinBERT neutral band shading (-0.05 to +0.05)
ax.axhspan(-0.05, 0.05, color='orange', alpha=0.2)
ax.text(
    df_sentiment_by_role.index[-12], 0.055,
    "FinBERT Neutral Range (-0.05 to +0.05)",
    fontsize=10, color='darkorange', style='italic'
)

# Get y-axis limits for label placement
ymin, ymax = ax.get_ylim()

# Add vertical lines for Bank Rate changes (red, top)
for _, row in df_bank_rate.iterrows():
    ax.axvline(row['date'], color='red', linestyle='--', alpha=0.7)
    ax.text(row['date'], ymax - 0.05 * (ymax - ymin), 'Bank Rate',
            rotation=90, fontsize=8, color='darkred', va='top', ha='right')

# Add vertical lines for Financial Stability events (blue, bottom)
for _, row in df_fin_stab.iterrows():
    ax.axvline(row['date'], color='blue', linestyle='--', alpha=0.7)
    ax.text(row['date'], ymin + 0.05 * (ymax - ymin), 'Fin Stability',
            rotation=90, fontsize=8, color='darkblue', va='bottom', ha='right')

# Add vertical lines for Monetary Policy events (green, center)
for _, row in df_mon_pol.iterrows():
    ax.axvline(row['date'], color='green', linestyle='--', alpha=0.7)
    ax.text(row['date'], ymin + 0.5 * (ymax - ymin), 'Mon Policy',
            rotation=90, fontsize=8, color='darkgreen', va='center', ha='right')

# X-axis setup
months = pd.date_range(start=start_date, end=end_date, freq='1MS')
ax.set_xticks(months)
ax.set_xticklabels([dt.strftime('%Y-%m') for dt in months], rotation=90)

# Final layout
ax.set_title("Monthly Average BoE Speech Sentiment Overall and by Role (2019 Onward) with bank events")
ax.set_xlabel("Month")
ax.set_ylabel("fb_compound Sentiment")
ax.axhline(0, color='black', linestyle='--', linewidth=1)
plt.legend(loc='upper left', bbox_to_anchor=(1.02, 1), borderaxespad=0)
ax.grid(False)
plt.tight_layout()
plt.show()
```

C.2 Model Limitations

<u>FinBERT Model:</u> The performance of natural language models such as FinBERT is significantly impacted by the training data used. The FinBERT model used in this analysis (yiyanghkust/finbert-tone) is pre-trained on a large corpus of earning call transcripts, corporate reports and analyst reports and further refined on 10,000 annotated sentences from analyst reports. BoE speeches, as part of central bank communications, have distinct attributes which may not be fully captured in the above training corpus, which may lead to inaccuracies. BoE can further improve performance by building a custom FinBERT fine-tuned to BoE communications.

C.3 Topic Keywords used for Labelling

Topic	Key Words
Inflation	inflation, CPI, price stability, interest rates, monetary policy, wage growth, cost of living, price pressure, price increases, consumer price index, monetary tightening, monetary easing, sticky prices, inflationary pressure, deflation
Economic Growth	GDP, FTSE, output, productivity, investment, unemployed, recession, economic, recovery, growth, demand, supply, expansion, consumption, slowdown, headwinds, spending, consumer confidence
Labour Market	employment, unemployment, wage growth, labour force, job creation, workforce, participation, hiring, vacancies, layoffs, tightening labour market, expanding labour market, labour productivity, furlough
Financial Market	stock markets, equities, FTSE, yields, guilts, FX, foreign exchange, investor sentiment, exchange rate, liquidity, market crash, boom, volatility, yield curve, risk, capital markets, credit spreads, bond market

D. Statistical Tests Output

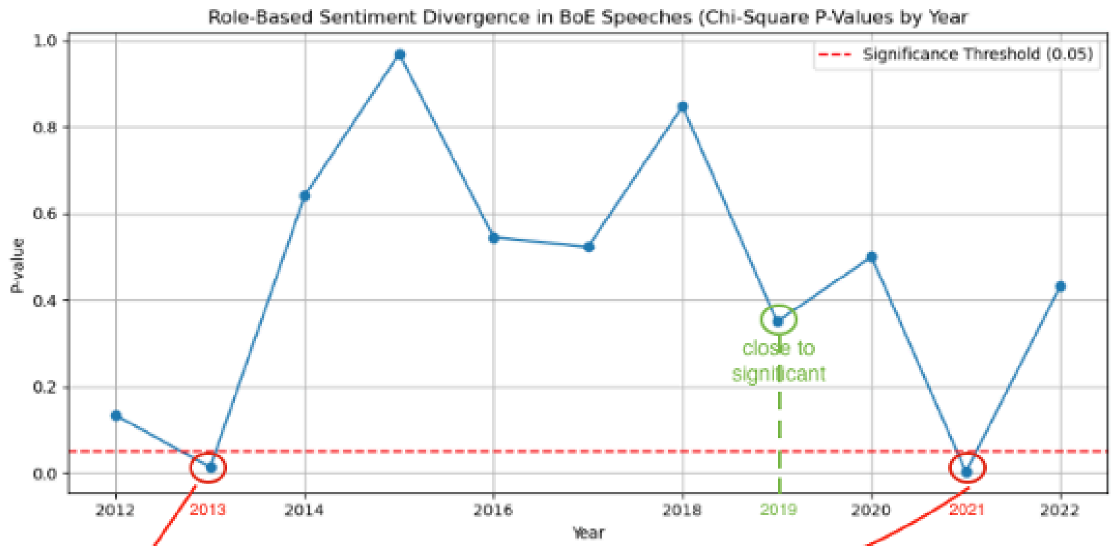
D.1 Chi-square, Granger Causality p values

D.1.1 Role-Based Sentiment Divergence in BoE Speeches

Role-Based Sentiment Divergence in BoE Speeches
(Chi-Square P-Values by Year)

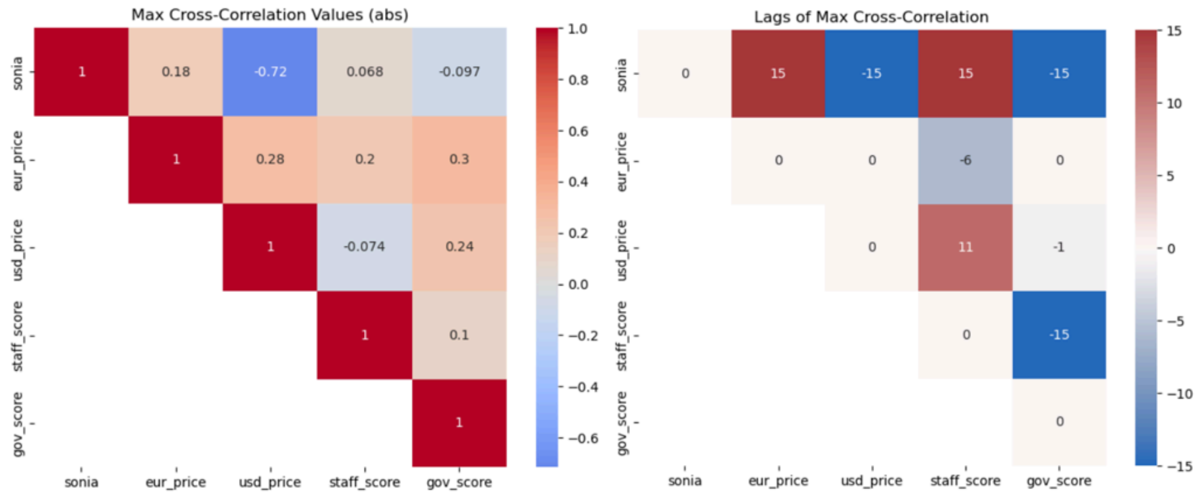
2105_JC 1it_EDA_2019_22Speeches_Events_Sentiment.ipynb

```python
# Ensure correct types
df_uk['date'] = pd.to_datetime(df_uk['date'])
df_uk['year'] = df_uk['date'].dt.year

# List to hold p-values
results = []

# Loop over each year where both Governor and Staff are present
for year, group in df_uk.groupby('year'):
    if group['is_gov'].nunique() == 2:
        contingency = pd.crosstab(group['is_gov'], group['fb_sentiment_custom']) \
                        .reindex(columns=['positive', 'neutral', 'negative'], fill_value=0)
        if contingency.shape == (2, 3):
            chi2, p, _, _ = chi2_contingency(contingency)
            results.append({'year': year, 'p_value': p})

# Convert results to DataFrame
df_pvals = pd.DataFrame(results)

# Plot the p-values
plt.figure(figsize=(10, 5))
plt.plot(df_pvals['year'], df_pvals['p_value'], marker='o')
plt.axhline(0.05, color='red', linestyle='--', label='Significance Threshold (0.05)')
plt.title("Role-Based Sentiment Divergence in BoE Speeches (Chi-Square P-Values by Year")
plt.xlabel("Year")
plt.ylabel("P-value")
plt.grid(True)
plt.legend()
plt.tight_layout()
plt.show()

# Optional: Print significant years
significant_years = df_pvals[df_pvals['p_value'] < 0.05]
print("\nYears with significant sentiment distribution difference between roles:")
print(significant_years)
```



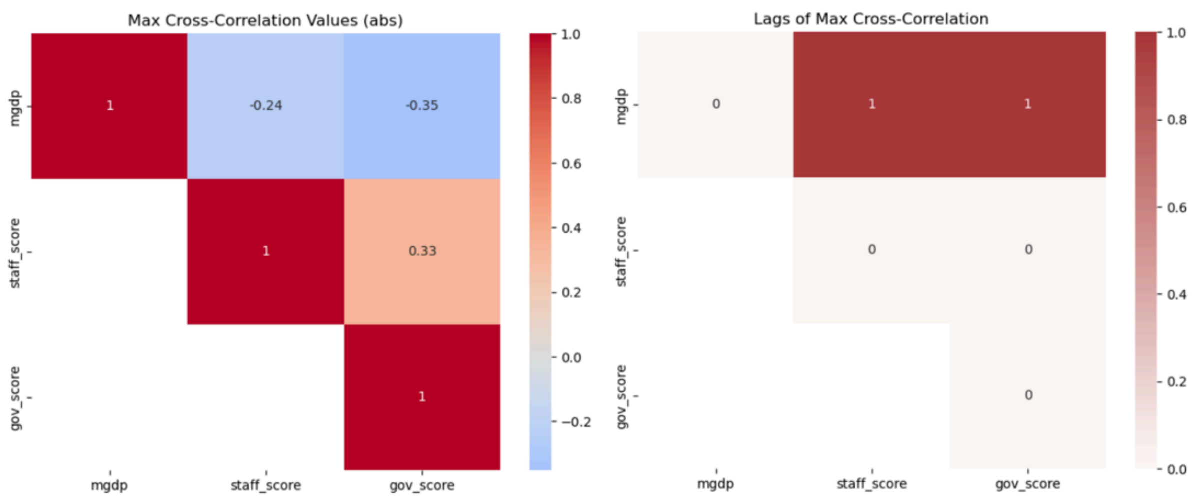

```
Years with significant sentiment distribution difference between roles:
   year   p_value
1  2013  0.014157
9  2021  0.003220
```

D.1.2 Cross correlation and matrix of lags of daily market indicators and sentiment scores (forward filled)



1805_AH_4it_EDA.ipynb

D.1.3 Cross correlation and matrix of lags of monthly GDP and monthly average sentiment scores (forward filled)



1805_AH_4it_EDA.ipynb

D.1.4 Code Example: Correlations Between Speech Sentiment and External Events

Code Example: Correlations Between Speech Sentiment and External Events

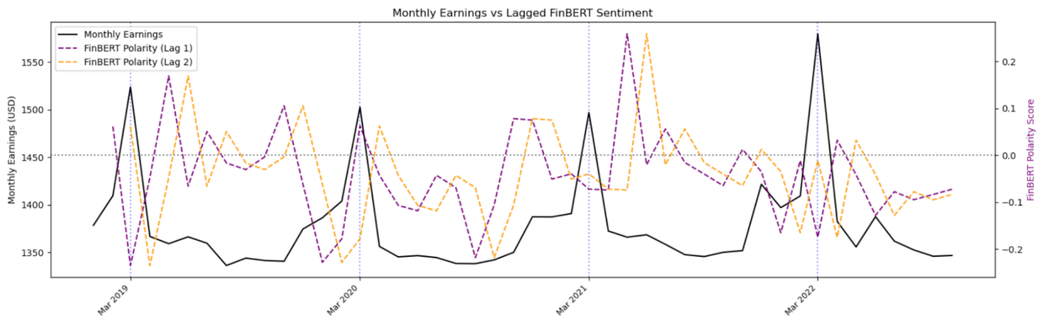
2105_JC_1it_EDA_2019_22Speeches_Events_Sentiment.ipynb

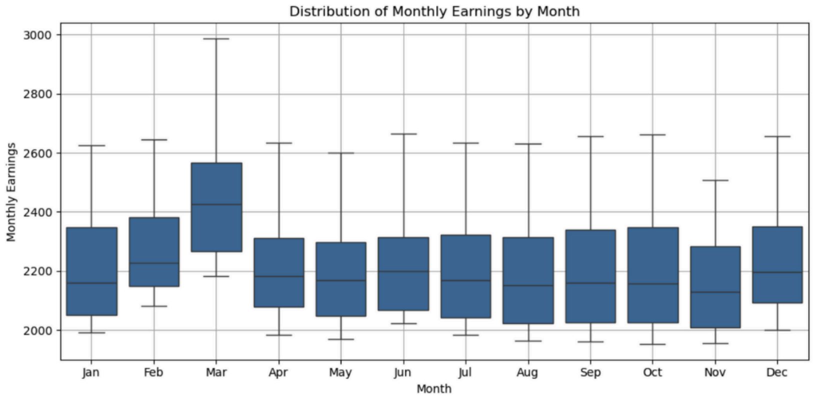
```python
# Step1 - Compute Correlation.
# Create a monthly date index to align with smoothed sentiment data
monthly_index = df_sentiment_smooth.index

# Utility function to create a binary event series
def make_event_series(event_dates, index):
    event_dates = pd.to_datetime(event_dates)
    event_months = event_dates.dt.to_period('M')
    index_months = pd.to_datetime(index).to_period('M')
    return index_months.isin(event_months).astype(int)

# Generate binary series for global and UK external events
external_event_df = pd.DataFrame(index=monthly_index)
external_event_df['global_event'] = make_event_series(df_ev_global_filtered['Estimated Start Date'], monthly_index)
external_event_df['uk_event'] = make_event_series(df_ev_uk_filtered['Estimated Start Date'], monthly_index)

#Step2 - Combine with Smoothed Sentiment Series
# Create sentiment DataFrame
sentiment_df = pd.DataFrame({
    'overall': df_sentiment_smooth,
    'inflation': df_inflation_smooth,
    'growth': df_growth_smooth,
    'labour': df_labour_smooth,
    'financial_markets': df_financial_markets_smooth
})

# Combine sentiment and event series
combined_external_df = pd.concat([sentiment_df, external_event_df], axis=1).dropna()

#Step3
# Pearson correlations between sentiment and external event indicators
external_correlations = combined_external_df.corr().loc[
    ['overall', 'inflation', 'growth', 'labour', 'financial_markets'],
    ['global_event', 'uk_event']
]

external_correlations.style.background_gradient(cmap='coolwarm').format("{:.2f}")
```

	global_event	uk_event
overall	-0.16	-0.16
inflation	0.13	-0.24
growth	-0.16	0.02
labour	-0.06	-0.23
financial_markets	0.17	0.07

E. Additional Findings

E.1 Wages

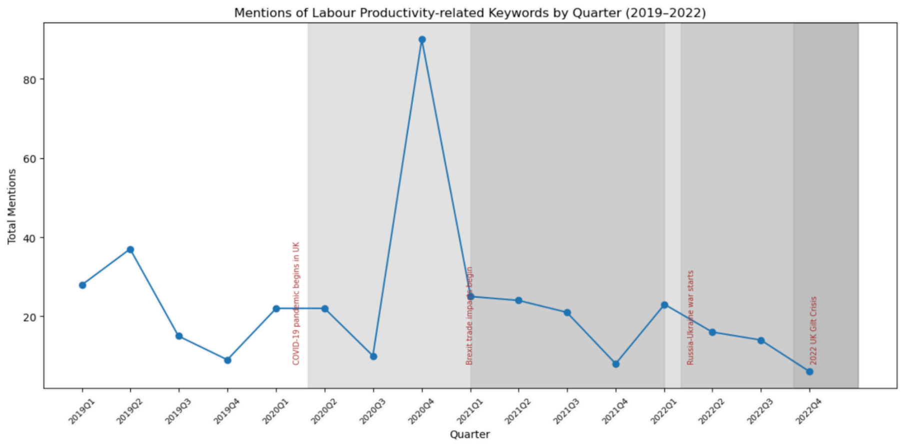




The observed fluctuations in wage-related data were not linked to macroeconomic events or sentiment in Bank of England (BoE) speeches. Instead, they reflect a seasonal cycle—likely influenced by annual pay reviews or fiscal year timing.

Implication: Short-term wage data shifts should not be over-interpreted as policy signals or sentiment changes without considering seasonality. Misattributing these changes may lead to incorrect conclusions.

E.2 Labour Productivity

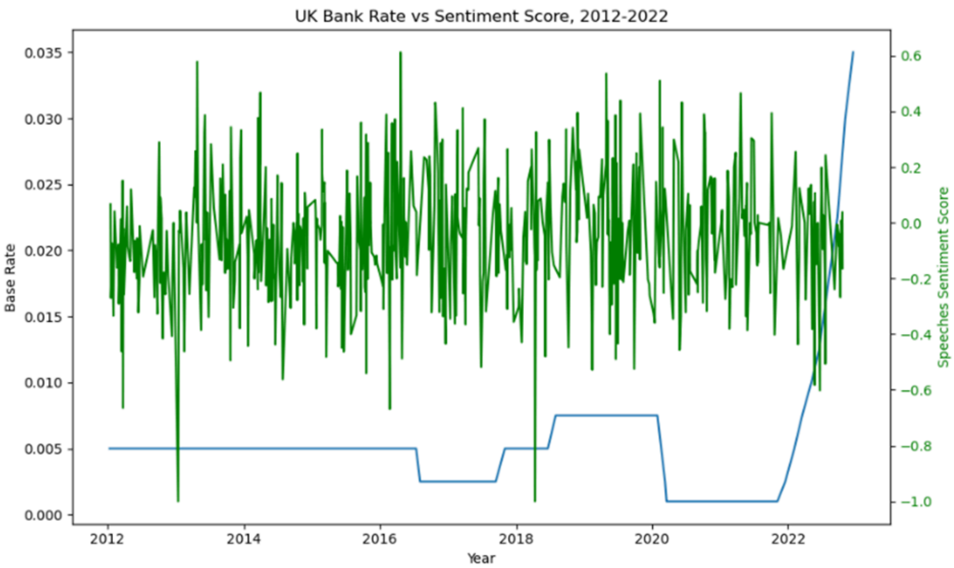


In late 2020, there was a sharp increase in the frequency of the term “productivity” in BoE speeches. This aligns with broader labour market shifts during the pandemic, particularly remote work and the focus on long-term productivity.

Despite the increased attention, sentiment scores remained stable. This suggests that a rise in keyword frequency can reflect changing priorities even if overall tone stays neutral. It underscores the value of keyword-based analysis in identifying evolving focus areas within policy discourse.

E.3 Bank Rate

Applying a Pearson correlation matrix between the UK Bank rate and overall FINBERT sentiment scores, there was a positive 17% correlation across an extended time period of 2012-2022. Given the majority of this time frame saw the Bank Rate at levels close to 0, the analysis was refined to 2019 onwards when rates began to rise. During this period, the relationship became inconsistent. In 2019 there was a negative 43% correlation, in 2020 the correlation is close to 0 and in 2022 there was a positive 31% correlation. It is likely this inconsistency was due to Bank rates reacting more so to external events such as COVID-19 Pandemic and Ukraine War than the sentiment of speeches.

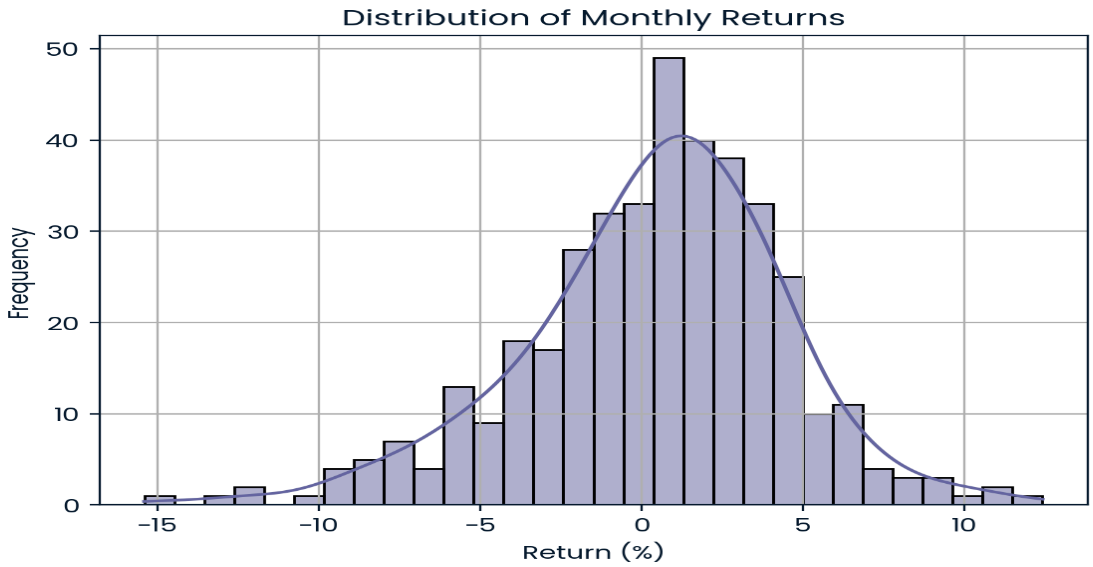


08_05_MS_bank_rate_3rd_iteration.ipnyb

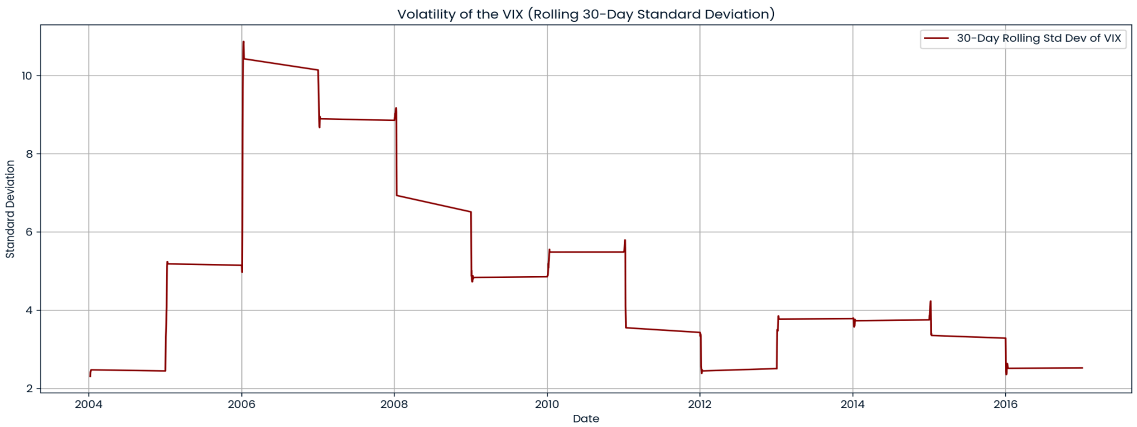
E.4 FTSE and FTSE VIX

Initially, the FTSE, VIX, and RPI from 1997 to 2022 were analysed to establish baseline trends. FTSE tracked broader economic cycles, VIX spiked during crises, and RPI showed a gradual rise with periodic inflationary shocks. We then narrowed the scope to 2019–2022 to capture heightened volatility driven by Brexit, COVID-19, the Ukraine war, and the UK Gilt crisis. This period revealed sharper dynamics: FTSE saw major swings, VIX remained persistently elevated, and RPI surged from mid-2021, reflecting mounting inflationary pressures linked to global disruptions and policy responses.

The distribution of monthly returns for the FTSE appears approximately normal but exhibits slight left skewness. This indicates that while most monthly returns cluster around small positive values, negative returns tend to be more extreme in magnitude. The mode sits just above 0%, suggesting that the FTSE typically delivers modest gains in a typical month. However, the distribution's long left tail reveals a higher frequency of large downturns compared to equivalent upward movements. This asymmetry reflects the market's downside risk profile and highlights the importance of accounting for tail risk when evaluating investment performance or constructing volatility-sensitive financial models.



The rolling 30-day standard deviation of the VIX shows that volatility is not constant, but rather clusters in periods of heightened financial uncertainty. Distinct spikes in the chart align with known periods of market stress, most notably the 2008 Global Financial Crisis and subsequent turbulence in 2011. After these peaks, volatility gradually declines and stabilises, reflecting a shift to a calmer market environment. This pattern reinforces the concept of volatility regimes, where investor sentiment and risk aversion vary across time. It also illustrates that the VIX, a proxy for market fear, can itself become volatile—making it a second-order indicator of instability that is especially relevant during systemic shocks.



F. References

FinBERT model: https://huggingface.co/yiyanghkust/finbert-tone