# Natural Language Processing for Economic and Financial Modelling


Maximilian Ahrens

New College

University of Oxford

Submitted for the degree of

*Doctor of Philosophy*

Trinity Term 2023

# Abstract

Over the past years, researchers have introduced various natural language processing (NLP) methods to the fields of economics and finance. It is time to take stock and systematically evaluate which NLP methods work best for the most common domain specific tasks, under the most common domain specific data characteristics, and to let those findings guide future research. We underline the importance of evaluating multimodal (text and numeric) data tasks, since many text datasets in economics and finance come accompanied by potentially relevant numeric metadata. Particularly in the field of modern monetary economics, multimodal data analysis is of vital importance. Ever since the global financial crisis, central bank communication has become a key monetary policy tool. The development of multimodal data analysis frameworks is pivotal for modern day monetary policy research, yet the integration of NLP methods into economic and financial modelling processes is far from being conclusively answered.

In this DPhil thesis, we introduce work towards establishing an NLP benchmark foundation for economics and finance that includes multimodal dataset evaluation tasks. Next, we introduce Bayesian Topic Regression (BTR). BTR is a multimodal NLP algorithm based on a supervised topic model that can jointly model text and numeric data. With BTR, we provide the research community with a method for more reliable causal inference with text data, in an identification framework commonly used in the economics and finance literature. BTR also demonstrates competitiveness in prediction tasks that rely on text and numeric data. Finally, we introduce a multimodal NLP framework for the application space of monetary policy analysis. We establish a new monetary policy shock series based on central bank speeches along three key macroeconomic measures: GDP growth, inflation, and unemployment. Based on empirical estimates on our newly constructed central bank communication dataset, our monetary news signals imply that news on macroeconomic outlooks in central bank communication can help explain equity and bond market volatility and tail risk. The news signals carry relevant information to which markets attend and react. We also derive a monetary news dispersion index, measuring the degree of alignment amongst different central bankers in their policy communication with the markets. Our findings suggests that more misaligned policy communication is associated with stronger market surprises at policy announcement time.

# Statement of Originality

This DPhil thesis and the work to which it refers are the results of my own efforts. I am furthermore the main author of all four of my experimental contribution chapters (Chapters 4-7), which have been joint research projects with various co-authors. I outline the individual contributions below. Any ideas, data, images, or text resulting from the work of others (whether published or unpublished) are fully identified as such within the work and attributed to their originator in the text, bibliography or in footnotes. All work has been conducted under the general supervision and guidance of Professor Stephen Roberts and Jan-Peter Calliess.

**Chapter 4:** The idea for the benchmark was jointly developed between Michael McMahon and myself. I designed the benchmarking framework and code implementation, conducted the experiments, and wrote the paper. Michael and I jointly evaluated the results. I am the first author of this paper.

**Chapter 5:** The idea was jointly developed between Julian Ashwin and myself (the two main authors). We developed the theoretical concept and all mathematical derivations jointly. Julian implemented the BTR model in *Julia* code, while I implemented all other reference models. Julian designed the synthetic data experiments. I designed the semi-synthetic and experimental data experiments. I am the first co-author of this paper.

**Chapter 6:** The idea has been jointly developed between myself, Michael McMahon, and Deniz Erdemlioglou. I designed and executed all experiments and wrote the code for all model implementations and experiments. The only parts not written and developed by myself are the continuous-time tests for volatility and tail risks. These are based on Deniz's and Xiye's contributions as well as on their previous papers. I am the first author of this paper.

**Chapter 7:** The idea has been jointly developed between Michael McMahon and myself. I design and executed all experiments and wrote the code for all model implementations and experiments. Michael and I jointly evaluated the results. I am the first author of this paper.

# Acknowledgements

I would like to thank my supervisors Professor Stephen Roberts and Jan-Peter Calliess for their invaluable support and guidance along the entirety of my DPhil journey. I wholeheartedly thank the the Economic and Social Research Council of the UK and the Oxford-Man Institute for supporting me with a DPhil scholarship. Particular thanks also goes to the Oxford-Man Institute and Professor Álvaro Cartea for financially supporting my endeavour to found and establish the NLP EcoFin/SoDaS conference series at the University of Oxford.

I am furthermore deeply indebted to Professor Michael McMahon for his fantastic mentorship, for his time and patience in all our meetings, for his unrelenting support, and for his unparalleled Irish humour making our collaborations tremendously enjoyable. I also want to thank all co-authors with whom I had the pleasure to collaborate over the past years.

A very special thanks goes to my friends Frances, Joel, and Salil for our innumerable discussions and wonderful moments during these years. You helped me to recover from all the lows during the journey and never gave up on me. I will never forget that. Sam, Marcus, and Giuseppe – you were the best flatmates one could ask for. Δάφνη, είμαι απεριόριστα ευγνώμων που οι ζωές μας διασταυρώθηκαν.

Finally, to my parents – Simone and Eckhard – and my grandparents – Helga and Hermann – without your unconditional support and belief in me, I would never have had the chance to pursue my academic dreams in the first place. You were my anchors when the seas got rough. This thesis is for you.

# Contents

# List of Figures

# List of Tables

# 1 Introduction

## 1.1 Relevance of NLP in Economics and Finance

In the past decades, macroeconomic and financial forecasts have regularly fallen short of explaining and predicting large fluctuations in economic activity and financial asset prices. Cases in point are the global financial crisis of 2007-08 or recent crypto-currency boom-and-bust cycles. Ever since the global financial crisis, we have also witnessed central banks' increased focus on policy communication (such as forward guidance and increased decision making transparency), as a tool to convey and implement monetary policy and to steer financial markets.

Stripped to their core, such examples reflect exchanges of information as well as economic and financial decision-making based on such information by a significant group of market participants and key financial market actors. While the exchange of information between individuals has long been established as an integral part of economic decision-making theory, standard financial and economic analysis has largely been focused on the exchange of quantitative information between agents - information that is usually available in form of tabular numeric data. However, a significant amount of informational exchange between participants in economies and markets is based on language (McCloskey and Klamer, 1995; McCloskey, 2016; Shiller, 2017). For example, information conveyed in natural language – say in news media, social media, or corporate reports – lets us measure objects that cannot easily be measured via traditional numeric data. We can think of measures for news sentiment (Shapiro et al., 2022), climate change action (Webersinke et al., 2022), or geopolitical events (Mueller and Rauh, 2018). Information from natural language often also allows us to capture higher frequencies of important economic and financial figures that are only available to us in rather low frequency via classic numeric data sources. For example, classic numerical macroeconomic data on labour market dynamics usually

comes in rather infrequent update intervals of months or quarters. Through natural language processing of online searches and social media messages one can construct much higher frequency approximations (up to real-time) for labour market indicators such as job loss, job search, and job postings (Antenucci et al., 2014). Overall, information conveyed in natural language can allow us to capture more relevant information and to capture this information in higher frequency compared to many classic numerical macroeconomic data sources. This in turn can help us model market agents' expectation formations and market dynamics more accurately, which is often critical for accurate economic and financial forecasting exercises. For example, information derived from natural language could be used to better explain market (in)attention dynamics in expectation formation processes with information rigidities (Coibion and Gorodnichenko, 2015), behavioural narrative formations (Shiller, 2017), or herding behaviour (Banerjee, 1992).

Due to its unstructured, textual data format, information conveyed in natural language has been inherently more difficult to analyse systematically and hence, has for a long time been neglected from (macro)economic and financial modelling. Over the last decade, however, advancements in natural language processing (NLP) methods have started to allow for ever more nuanced and precise semantic and sentiment assessments of text-based datasets in economics and finance (Xing et al., 2018; Gentzkow et al., 2019). Such methodological progress as well as ever-increasing volumes of structured and unstructured data sources have spawned a still relatively young but rapidly growing research community in the intersection of natural language processing and economic and financial modelling. In the field of computer science and engineering for instance, ACL[1] and EMNLP[2], two of the leading conferences in the field of computational linguistics, have hosted the ECONLP workshop on Economics and Natural Language Processing since 2018. They state that the creation of such an intersectional platform "[...] *addresses the increasing relevance of natural language processing for [the] regional, national and international economy, both in terms of already launched language technology products and systems, as well as new methodologies and techniques emerging in interaction with the paradigm of computational social science.*"[3]. Likewise, in economics and finance, leading academics such as Shiller (2017), as well as major central banks such as the US Federal Reserve Bank (Fed) (Fed, 2019), the European Central Bank (ECB) (B. Coeuré, 2017), and the Bank of England (BoE) (Bholat et al.,

[1]Meeting of the Association for Computational Linguistics
[2]Conference on Empirical Methods in Natural Language Processing
[3]aclweb.org/portal/content/econlp-2018-1st-workshop-economics-and-natural-language-processing-acl-2018

2015), have stressed the importance of big data and textual analysis to enhance the accuracy of economic and financial analyses. Finally, as a product of my own research efforts during my DPhil studies and my collaborations with the Oxford-Man Institute, I founded, organised, and chaired the international research conference on Natural Language Processing for Social Data Science (NLP SoDaS). Thanks to the generous support by the Oxford-Man Institute, we have been able to establish a research conference at the intersection of NLP and economic and financial modelling as well as the wider sphere of computational social science. Inaugurated in 2022, we hope the conference will continue to underline the relevance of this interdisciplinary research area also in the years to come.

## 1.2 Research Contributions

Economic and financial researchers have introduced a plethora of text analysis methods into their domain over the past years - often inspired by the tremendous technological breakthroughs in the fields of computational natural language processing and machine learning (Blei et al., 2003; Mikolov et al., 2013a; Vaswani et al., 2017; Devlin et al., 2018; OpenAI, 2022). Most of those methods, however, were not designed to handle data characteristics oftentimes present in economics and finance. Datasets in these domains usually feature fewer observations, relatively long document lengths, time-series dynamics, and endogenous relationships between text and numbers. The mainstream NLP community has particularly thrived through the availability of common benchmark datasets and tasks. There is a lack of such NLP benchmark in economics and finance (Ash et al., 2021).

With this thesis, we aim to introduce the foundational work towards a systematic NLP benchmark in these domains. In particular, we underline the importance of evaluating multimodal (text and numeric data) tasks, since many text datasets in economics and finance are accompanied by potentially relevant numeric metadata. For instance, economic reports and financial statements are usually written in the context of current market conditions. Without the economic and financial context that is often encoded in numeric data, text passages can be ambiguous in their meaning. What is needed is a wider benchmark attempt, comparing the classes of NLP models that have been introduced into and adopted by the economics and finance domain over the past years and decades. It is important to take stock, systematically evaluate, and produce visibility

about which NLP models work best under which (multimodal) dataset characteristics in these fields. [Chapter 4]

Based on the identified needs for improved multimodal NLP modelling in economics and finance, we develop the Bayesian Topic Regression (BTR) model. BTR is a multimodal NLP algorithm based on a supervised topic model that can jointly model text and numeric data. We show that our BTR model respects the Frisch-Waugh-Lovell theorem - an important condition to make progress towards causal inference with observational text data. Causal inference with text data is a rather nascent research field which has witnessed important research contributions over the past years. Furthermore, we demonstrate that our model is also competitive in prediction tasks that rely on text and numeric data. [Chapter 5]

Finally, we develop a multimodal NLP modelling framework for the application space of monetary policy communication analysis. As mentioned previously in the introduction, central bank communication has become a major monetary policy making tool ever since the global financial crisis. However, theoretical and empirical work is still far from having a conclusive answer on, for example, the impact and effectiveness of central bank communication to steer financial markets. We therefore establish a new monetary policy shock series based on central bank speeches along three key macroeconomic measures: GDP growth, inflation, and unemployment. We focus on the US Fed and its members of the Federal Open Market Committee (FOMC) - the Fed's principal decision-making body for monetary policy.

Based on empirical estimates of our newly constructed FOMC member speech dataset, our monetary news shock series implies that news on macroeconomic outlooks in central bank communication can help explain volatility and tail risk both in equity and bond markets. Speech-implied news signals seem to carry relevant information to which markets react. Furthermore, we introduce a news dispersion index, measuring the degree of alignment amongst different central bankers and their communication with the markets. Our findings suggest that more misaligned (or 'cacophonous') policy communication in the build-up to FOMC meetings is associated with stronger subsequent market surprises at FOMC policy announcement time. Such findings underpin the importance of analysing the continuous communication process of

central banks with the markets - and speeches are a main communication element in this. With our multimodal NLP framework, we aim to facilitate further empirical research on central bank communication and other multimodal NLP modelling tasks in economics and finance. [Chapter 6 and 7]

Published papers based on these contributions are:

1. *Bayesian Topic Regression*, EMNLP, (Ahrens et al., 2021)
2. *Extracting Economic Signals from Central Bank Speeches*, EMNLP, (Ahrens and McMahon, 2021)
3. *Mind Your Language: Market Responses to Central Bank Speeches*, SSRN (*revise & resubmit* at Journal of Econometrics), (Ahrens et al., 2023)

## 1.3 Thesis Structure

This thesis is structured as follows: Chapter 2 introduces the main methodological foundations of NLP and the related background literature. Chapter 3 outlines the foundational underpinnings for NLP in monetary policy analysis. What then follows are four experimental contribution chapters: Chapter 4 introduces foundational work towards a comprehensive NLP benchmark for the domains of economics and finance. Chapter 5 develops a novel multimodal supervised topic model for prediction and causal inference. Chapter 6 develops a multimodal NLP modelling framework for the application space of monetary policy communication analysis. Chapter 7 applies the multimodal NLP framework to measure divergence in central bankers' speeches and its impact on market participants' expectation formations at FOMC announcement dates. Furthermore, it creates a monetary policy news index according to three key macroeconomic dimensions - GDP growth, inflation, and unemployment. Chapter 8 concludes this thesis and points towards areas for future work.

# 2

# Background - NLP Methods

This chapter introduces the required terminology and methodological foundations in NLP relevant for this thesis. The end of this chapter outlines some of the key challenges and gaps in the literature regarding the application of NLP methods in economics and finance.

**Foundational terminology for NLP:** To work with text in quantitative analysis and in machine learning algorithms, we need to first define a common terminology of the elements of text and how to represent those elements in a machine readable format. Let there be a corpus $\boldsymbol{\mathcal{T}}$ that contains all $D$ text documents $\boldsymbol{\tau}_d$, $d \in \{1, ..., D\}$, such that $\boldsymbol{\mathcal{T}} = [\boldsymbol{\tau}_1, ...\boldsymbol{\tau}_D]^\top$. Each document $\boldsymbol{\tau}_d$ itself consists of $N_d$ tokens $\tau_{n,d}, n \in \{1, ..., N_d\}$, $d \in \{1, ..., D\}$ such that $\boldsymbol{\tau}_d = [\tau_{1,d}, ..., \tau_{N_d,d}]$. A token can be a single word, a group of words, or sub-word elements. The different tokenization approaches will be covered in this chapter. Finally, let a vocabulary $\boldsymbol{v}$ be the set of all of the $U$ many unique token IDs $v_u$, $u \in \{1, ..., U\}$ in corpus $\boldsymbol{\mathcal{T}}$, such that $\boldsymbol{v} = \{v_1, ...v_U\}$.

## 2.1 Preprocessing

### 2.1.1 Tokenization

A first text processing step is to decide on how to break a raw text string into its individual elements - commonly referred to as *tokens*. A multitude of different approaches exist, which would make an exhaustive and in-depth analysis of the field go beyond the scope of this thesis. Mielke et al. (2021) provide a recent and detailed survey paper about the evolution of tokenization for NLP models. We will outline the most important foundations, developments, limitations, and challenges regarding tokenization.

Tokenization represents the task of segmenting a raw text string into individual units (tokens) that become the input for language models. Until recently, the most common approach to tokenization has been to perform segmentation on the word-level. This approach might also be the most intuitive to a human. In this case, a token is equivalent to a word.

This would result in breaking a text string such as 'I live in New York' into its tokenized version which would be [I, live, in, New, York], consisting of five unique tokens. However, we can also think of representing the text according to its n-grams, which are the result of treating $n$ many words as one token. To select which words to combine as n-grams, one can count all token occurrences in a corpus and assess whether a certain token tends to frequently co-occur next to other tokens or whether its neighbouring words are rather interchangeable. The decision on when to create an n-gram and when to stick with a unigram ($n = 1$) is usually based on some form of a relative frequency threshold. For more details see Daniel and Martin (2021). As an intuitive example, we might find that 'York' tends to be predominantly preceded by 'New' in our context, so that we might prefer to use the n-gram token 'New_York' whenever we see the token 'New' and 'York' back to back in a string. On the other hand, 'in' might only occasionally be preceded by 'live', so we rather use a separate token for 'live' and for 'in' than the bi-gram 'live_in'. In this n-gram example, our tokenized example sentence would be [I, live, in, New_York], and would now consist of only four unique tokens.

It is worth mentioning that other preprocessing steps might be conducted even before the tokenization. As any preprocessing step, they depend on the individual language, but might contain operations such as lower casing, and deleting *stopwords* and numbers or replacing them with special characters. Also operations such as stemming and lemmatization have often been applied to further condense the list of unique tokens in the vocabulary. Stemming reduces words to their *stem* by cutting off ends of of words such as affixes (Manning et al., 2008). For example, one of the most commonly used stemming algorithms, the Porter stemmer (Van Rijsbergen et al., 1980), would take an input sequence such as [*economists are studying economics quite economically until they quit university. i am studying the universe.*] and would return the following stemmed version [*economist ar studi quit economi until thei quit univers. i am studi the univers.*]. We achieve a reduction in overall token size, however, it comes at the cost of losing potentially important contextual information as we can no longer differentiate between, say,

'economics' and 'economical', between 'quit' and 'quite', or between 'universe' and 'university'.

Lemmatization on the other hand attempts to remove inflectional endings only through the use of a vocabulary and morphological analyses of the respective words (Manning et al., 2008). This reduces a word to what is referred to as a *lemma*. Applying lemmatization to our previous example should lead us to [*economist be studying quite economically until they quit university. i be studying universe*]. We might achieve overall less token reduction, but preserve more potentially important meaning.

Whilst tokenization by words or word n-grams might be the most intuitive approach, this method can lead to issues especially when dealing with very large text corpora. Word-level segmentation usually requires the generation of very large vocabularies to be able to minimize the amount of unknown words during inference, also called out-of-vocabulary (OOV) tokens.

Large vocabulary sizes lead to exceedingly large token embedding matrices and in turn to substantially increased memory and computational costs. Therefore, more recently, tokenization has moved away from word or word n-gram based segmentation to subword, character or byte level approaches. Such methods might lead to non-typographically and non-linguistically motivated tokens (Mielke et al., 2021), however it has become the de facto standard tokenization process for modern language models such as algorithms of the BERT (Devlin et al., 2018), T5 (Raffel et al., 2020) or GPT families (Liu et al., 2019; Radford et al., 2019; Brown et al., 2020). Data-driven algorithms such as Byte Pair Encoding (BPE) (Sennrich et al., 2016), WordPiece (Schuster and Kaisuke, 2012; Wu et al., 2016), Unigram (Kudo, 2018), or SentencePiece (Kudo and Richardson, 2018) segment the raw text based on frequencies of sub-word, character, or byte pairs.

Those subword tokenization approaches operate on the premise that frequently occurring words will be encoded on the word-level. Rare words however, will be broken down into sub-word pieces. The idea is that with a sufficiently large yet overall limited amount of those sub-word tokens, we can represent almost all words occurring in a corpus. For instance, the potentially relatively rare word 'economically' might get broken into more frequently occurring subword elements 'economic' and 'ally'. Similarly, the word 'logically' might be decomposed into 'logic' and 'ally'. The subword-level tokens obtain their own representation. However, through the combination of them we can still aim to be able to recover the original meaning of,

say, 'economically' and 'logically'. This subword-level decomposition method allows models to operate with a relatively small subword-level vocabulary size, which nevertheless allows them, through sub-word token combination, to still represent most word-level meanings. Furthermore, this approach allows models to deal with words they have not seen before in training, as these unknown words can be decomposed into known subword elements.

The key technical differences between BPE, WordPiece, and Unigram lie in the different merging rules, i.e. according to which rules a word might get broken into subword elements. For further reference to the subword tokenization algorithms, see for instance (Mielke et al., 2021). SentencePiece is in itself not a tokenization algorithm. It rather is based on either BPE or Unigram, however, it does not treat whitespace as necessary word boundaries. This allows it to operate on languages that don't use whitespaces as their general word boundaries such as Japanese, Chinese, or Thai (Kudo and Richardson, 2018).

It remains important to note, that those latest subword-level methods are not without their limitations either. Cao and Rimell (2021) point out that those tokenizers operate by picking the one-best tokenization they identify based on their algorithmic process, ignoring tokenization uncertainty and thereby alternative segmentation processes. The researchers argue that evaluating on the marginal likelihood over all tokenization possibilities can yield superior out-of-domain performance. This critique falls into a similar canon of recent research, that proposes token-free end-to-end natural language models (Clark et al., 2022; Xue et al., 2022; Tay et al., 2021). Those papers argue that tokenization on word- or subword-level, as an independent preprocessing step to create the input for the subsequent language model, is sub-optimal for downstream NLP tasks both in high and low resource languages, and is rather a relic of legacy NLP processes. However, it is important to mention that virtually all state-of-the-art language models, say models in the BERT, T5, and GPT family, still work with separate tokenizers that preprocess the model input.

To point out some of the limitations of relying on separate tokenization processes, even in high resource languages, BERT's (Devlin et al., 2018) performance is sensitive to both naturally occurring corruptions in input text (e.g. typos) as well as to adversarial designed corruptions (Pruthi et al., 2019; Sun et al., 2020). The spelling sensitive tokenization process, plays a pivotal factor in this. Furthermore, there exist a multitude of low-resource languages

that, for instance, are based on substantially richer and different morphological characteristics than Germanic or Romance languages (Clark et al., 2022). Token-free end-to-end language models can theoretically overcome these issues (Clark et al., 2022). Clark et al. (2022) propose a tokenization-free, vocabulary-free deep encoder for language modelling. Xue et al. (2022) and Tay et al. (2021) both develop a tokenization-free version of the multilingual mT5 model (Raffel et al., 2020). These models indicate a promising future research path, as their initial results suggest more robust model performance, for example, to noise and spelling mistakes. Furthermore, they achieve comparable performance results on various NLP datasets and benchmark tests whilst being more computationally efficient (Tay et al., 2021).

It remains to be pointed point out though, that there does not yet exist a one-size-fits-all solution across languages, domain application, and downstream tasks. Finally, most of these tokenization method innovations were published in the last stage of this DPhil thesis. Modern state-of-the-art methods are still using (sub)word tokenization to this day. Topic modelling approaches have predominantly followed word or word n-gram level tokenization methods. We followed this approach and believe that this is not detrimental to our overall research findings.

### 2.1.2 Token-to-ID Mapping

After the tokenization step, the next process is the mapping of tokens to a unique ID, and then in turn to define or learn a vector representation for each of the IDs.

Each token gets mapped to to its position in the vocabulary of unique tokens. As mentioned previously, the vocabulary size can have substantial impact on the computational costs as it defines (and often restricts) the text feature space. Let us take our previous example sentence 'I live in New York' and let us add a second sentence 'I work in New York too'. Let our vocabulary $\boldsymbol{v}$, as defined previously, be the set of all unique tokens in our corpus. Using word n-gram level tokenization, we get $\boldsymbol{v}$ = {I, live, in, New_York, work, too} (let's assume New York transpired as a bi-gram out of our tokenization process). Our vocabulary has a size of 6 unique tokens, each of them we can map to a corresponding ID = $[1, 2, 3, 4, 5, 6]$. Now that each token can be uniquely mapped to an ID in our vocabulary, we have to face the decision on how we want to represent the token in a machine readable form so that we can conduct mathematical operations with it. The representations can range from simple word frequency counts to elaborate word

embedding representations extracted from a hidden layer in a large language model (LLM).

## 2.2 Count-Based Token Representations and NLP Models

The following sections outline the most commonly used approaches of how to define and obtain token representations, with a particular focus on the application space of NLP in economics and finance. The sections also briefly outline the advantages and disadvantages of each approach.

### 2.2.1 Dictionary Models

During the early stages of textual analysis in economics and finance, researchers used hand-coded dictionaries such as the Harvard General Inquirer dictionary (Stone et al., 1966) to categorise words into positive and negative classes. However, the Harvard General Inquirer dictionary assesses words according to their general meaning in the English language. In economic and financial contexts, words can have quite a different meaning and sentiment from the original use of the word. For example, the word 'liability' might have a negative general connotation but would most likely be considered neutral in a financial context. Various researchers have painstakingly manually adopted such dictionaries to economic and financial contexts.

Probably the most commonly used financial dictionary of this sort is the one by Loughran and McDonald (2011) (LM), which to this day, enjoys wide application among researchers and practitioners in economics and finance. The LM dictionary provides word lists along seven categories: negative, positive, uncertain, litigious, strong modal, weak modal, and constraining. In application, researchers and practitioners often focus only on the first four of these categories (Das et al., 2022). This representation of the text creates a very small feature set - one feature for each category $c$ - and is therefore very easy and cost efficient to use. Each document is then represented through a $c$-dimensional vector, each entry representing the relative prevalence of that feature in the document. However, despite being manually tailored towards financial contexts, deployment of the LM - or any dictionary of that sort - should still be considered thoughtfully. The scope of contexts in economics and finance is still vast. Vernacular in the context of, say, bankruptcy assessments in corporate finance might still markedly differ from language that central bankers use to describe their outlook on the economy. No single dictionary approach can provide a silver bullet for all economic and financial domain applications.

Regarding the widely used LM dictionary, words such as 'hyperinflation', 'crash', 'blackout', or 'subdued' are not even marked as pertaining to any of the dictionary categories. The list of potentially problematic hand-coded word classifications is long.

One immediate solution can be more data-driven dictionary methods. Most recently, Li et al. (2021) used Word2Vec word-embeddings (Mikolov et al., 2013b) (explained in later sections) to create dictionary lists along five corporate cultural values of innovation, integrity, quality, respect, and teamwork. They first train a neural network to learn word embeddings based on over 200,000 earnings call transcripts. Subsequently, they fill each dictionary list with words that are semantically closest (measured via cosine similarity between word-embeddings) to the above mentioned category 'value words'. The authors show that their word embedding based dictionary method creates measures for corporate culture that correlate with business outcomes such as operational efficiency, risk-taking, earnings management, executive compensation design, and firm value.

Das et al. (2022) take a similar methodological approach intended as a data-driven alternative to the LM dictionary in finance. Their FinLex model is based on the FastText word embeddings (Bojanowski et al., 2017). They re-create four LM categories (negative, positive, uncertain, litigious) by using these category words as seeds and generating word lists based on cosine similarity. Moreover, they add new lexica around the antonym pairs 'positive/negative', 'safe/risk', 'certainty/uncertainty', 'fair/unfair'. They benchmark their FinLex and Finlex+ model against the LM approach across three diverse datasets: Financial Phrase Bank[1], Disaster Tweets[2], and Reddit News[3]. Whilst the creation of the FinLex based lexica requires no manual work and can easily be created for any type of seed words, this approach does not markedly outperform the LM dictionary. Boukes et al. (2020) and van Atteveldt et al. (2021) find that dictionary model performance is often close to chance and inter-dictionary consistency is low. Such findings suggest that it might be time to default to more performant NLP model alternatives. The NLP model benchmark in Chapter 4 features a dedicated assessment of the performance in sentiment analysis of the workhorse dictionary methods by Loughran and Mcdonald (2011) against data-driven machine learning methods.

[1] https://www.kaggle.com/datasets/ankurzing/sentiment-analysis-for-financial-news
[2] https://www.kaggle.com/datasets/vstepanenko/disaster-tweets
[3] https://www.kaggle.com/datasets/aaron7sun/stocknews

### 2.2.2 Word Count Models

Word count regression or word count classification approaches are closely related to data-driven dictionary methods. But instead of only considering words from a manually pre-defined word list, each word in the corpus is being considered as carrying potentially relevant information.

#### 2.2.2.1 Document-Term-Matrix, Bag-of-Words, and Cosine Similarity

Usually, the first step is to apply a frequency measure on the word or n-gram occurrences of the corpus. This creates a document-term matrix (DTM) in which each row represents a document, each columns represents a unique token of the vocabulary. We can now represent the entire corpus in this matrix of token frequencies. Let us revisit the previous two sample sentences $\boldsymbol{\tau}_1$: 'I live in New York' and $\boldsymbol{\tau}_2$: 'I work in New York too'. Let us also add two more example sentences. $\boldsymbol{\tau}_3$:'Apples often come in green or red' and $\boldsymbol{\tau}_4$:'Bananas often come in green or yellow'. Translating those four sentences into a token-frequency based DTM, yields us a sparse matrix as depicted in Table 2.1.

Table 2.1: Document-term-matrix with token frequency counts for the example sentences

| | I | live | in | New_York | work | too | apples | often | come | green | or | red | bananas | yellow |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\boldsymbol{\tau}_1$ | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $\boldsymbol{\tau}_2$ | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $\boldsymbol{\tau}_3$ | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| $\boldsymbol{\tau}_4$ | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 |

From Table 2.1, we can also see that of all possible document vector pairs, the pairs ($\boldsymbol{\tau}_1$, $\boldsymbol{\tau}_2$) and ($\boldsymbol{\tau}_3$, $\boldsymbol{\tau}_4$) are the most similar ones to one another. In such a DTM creation procedure, documents that share the same tokens - as well as similar frequencies of those tokens - end up with more similar document vectors. It is the frequency of token occurrences - not the order - that matters in this representation approach. Hence also the name *bag-of-words* that is used for representation methods which ignore the word order. More formally, we can measure vector similarity for example via cosine similarity (cs),

$$\text{cs}(\boldsymbol{d}_1, \boldsymbol{d}_2) = \frac{\boldsymbol{d}_1 \cdot \boldsymbol{d}_2}{||\boldsymbol{d}_1|| ||\boldsymbol{d}_2||}, \tag{2.1}$$

where we calculate the dot product between the two vectors in the nominator, and the product

of the vectors' magnitude in the denominator. Cosine similarity is a widely used similarity measure in NLP.

We can now feed each row of the DTM as a feature set for a data point into a machine learning algorithm, the simplest examples being a linear regression for regression tasks or a logistic regression for classification. Our features are now represented in the DTM which is our data matrix $\boldsymbol{X} \in \mathbb{R}^{D \times U}$. If we have a label for each document $\boldsymbol{y} \in \mathbb{R}^{D \times 1}$ (same is true for the multilabel case), we can start learning the model parameters, assuming we train a parametric model like a word (logistic) regression or other parametric machine learning models. In general, for any function with parameters $\boldsymbol{\omega}$ that maps our data matrix $\boldsymbol{X}$ to the target variables $\boldsymbol{y}$ we get $\boldsymbol{y} = f(\boldsymbol{X}; \boldsymbol{\omega})$. In the linear case, we have $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\omega}$ for regression and $\boldsymbol{y} = \sigma(\boldsymbol{X}\boldsymbol{\omega})$ for logistic regression. Here, $\sigma$ represents the logistic function, $\boldsymbol{\omega} \in \mathbb{R}^{U \times 1}$ represents the trainable (logistic) regression parameters.

#### 2.2.2.2 Relevance Weighting

A commonly applied modification to the DTM is a transformation of the raw frequency counts into relevance-weighted counts. One of the most popular of such approaches is the *term-frequency inverse-document-frequency* (tf-idf) weighting. Slightly different implementation rules exist. However, the overall approach is the same: increase the importance of words that occur in high frequency (term-frequency), but that only occur in a few documents (inverse-document-frequency). The tf-idf weighting therefore adjust for the linguistic pattern that some words generally occur more frequently than others. Term-frequency (tf) is the relative frequency of token $v_u$ within document $d$, such that

$$\text{tf}(v_{u,d}) = \frac{\sum_n^{N_d} \mathbb{1}(\tau_{n,d} = v_u)}{N_d}. \tag{2.2}$$

Here, $\mathbb{1}(\tau_{n,d} = v_u)$ is the indicator function whether token $\tau_{n,d}$ is equal to the unique token $v_u$ in the vocabulary. The nominator therefore represents the raw count of the occurrences of unique token $v_u$ in document $d$. Alternatives exist, for instance the raw count term-frequency per document $\text{tf}_{\text{raw}}(v_{u,d}) = \sum_n^{N_d} \mathbb{1}(\tau_{n,d} = v_u)$ or a logarithmic term-frequency $\text{tf}_{\log}(v_{u,d}) = \log(1 + \sum_n^{N_d} \mathbb{1}(\tau_{n,d} = v_u))$. The inverse-document-frequency (idf) assesses how common a token is across all documents, which in turn reflects how much informational value the token carries.

As with the tf, several idf implementation variants exist. It is usually a logarithmically scaled inverse ratio of the number of documents in which the token $v_u$ occurs over the total number of $D$ documents in corpus $\boldsymbol{T}$, such that

$$\text{idf}(v_u, \boldsymbol{T}) = \log \frac{D}{\sum_d^D \mathbb{1}[\sum_n^{N_d} \mathbb{1}(\tau_{n,d} = v_u) \neq 0]}. \tag{2.3}$$

The tf-idf weighting multiplicatively combines the tf and the idf measure, where we define

$$\text{tfidf}(v_u, \boldsymbol{\tau}_d, \boldsymbol{T}) = \text{tf}(v_{u,d}) \cdot \text{idf}(v_u, \boldsymbol{T}). \tag{2.4}$$

In our example DTM (see Table 2.1), token frequencies don't exceed one per document. This is of course simply due to our simplistic document constructions. We can see that the token 'in' has the same frequency in all documents. It therefore carries no relevant information to differentiate between documents. We might consider words that appear across (almost) all documents in similarly high frequency as *stop words* which carry minimal informational value. We might opt to delete these tokens from our vocabulary. In many practical applications, tf-idf weightings are being applied to the training corpus and only a certain top percentage share of the most informative words would be kept in the vocabulary, which can markedly shrink the size of the data matrix.

#### 2.2.2.3 Regularization

We might also apply a penalty term such as $L^1$ norm (lasso), $L^2$ norm (ridge), or a linear combination of both (elastic net) to regularize the number of features. This can be particularly important in economics or finance applications where a linear model is applied and the number of features $U$ is larger than the number of observations $D$. For example, in a setting of $U > D$, for a linear regression without regularization we would have to estimate $\boldsymbol{\omega} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$ where we cannot invert $\boldsymbol{X}^\top \boldsymbol{X}$ since it would no longer be full rank (due to arising multi-colinearity problems) and therefore an estimate for the regression parameters does not exist. As an example, for the linear regression with an elastic net regularization, we change the loss function for the

regression parameter estimation from the least squares loss function

$$\widehat{\boldsymbol{\omega}}_{ls} = \arg\min_{\boldsymbol{\omega}}(||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\omega}||^2) \tag{2.5}$$

to the regularized least squares loss function

$$\widehat{\boldsymbol{\omega}}_{rls} = \arg\min_{\boldsymbol{\omega}}(||\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\omega}||^2 + \lambda_1||\boldsymbol{\omega}||_1 + \lambda_2||\boldsymbol{\omega}||_2^2), \tag{2.6}$$

where $\lambda_1$ is the $L^1$ norm penalty term (lasso) and $\lambda_2$ is the $L^2$ norm penalty term (ridge). $L^1$ and $L^2$ norms are conventionally defined as $L^1 : ||\boldsymbol{\omega}||_1 = \sum_u^U |\omega|$ and $L^2 : ||\boldsymbol{\omega}||_2 = \sum_u^U |\omega|^2 = \sqrt{\omega^2 + ... + \omega^2}$.

Such word count methods are relatively easy to implement and they often show quite competitive performances in economic and financial modelling tasks such as sentiment analysis and sequence classification (see Chapter 4). However, they provide only a very limited reduction in the feature space. And despite the relative simplicity of such models, interpretability is not straightforward. For example, when using regularization techniques, such as lasso, the algorithm might select token features that are strongly partially correlated with the target variable whilst dropping other features that have similarly strong partial correlations with the target variables, but that are also strongly correlated with the already selected feature. Whilst this approach can yield strong predictive performance, interpretation of the 'important' features has to be done with extreme care and oftentimes is impossible to be done rigorously at all. Ridge regularization can remedy this problem of handling multicolinearity in the feature space, as it does not set feature coefficients to zero. Ridge therefore does not perform feature selection. Whilst ridge does not drop highly correlated features, it also does not shrink the feature space at all, which again does not help the case of model simplicity.

Finally, as mentioned previously, all approaches that are based on such a DTM input, are bag-of-words approaches and therefore neglect the word order. To what extent the neglect of word order information affects the predictive task performance often depends on the dataset and domain specific use case. In our model benchmark in Chapter 4, we show that at least for tasks such as sentiment analysis and sequence classification, bag-of-words assumptions haven proven not to be too detrimental for model performance.

### 2.2.3 Topic Models

Factor and topic models represent an alternative approach to extracting relevant information from the DTM whilst substantially reducing the feature space. Topic models can be interpreted as a form of factor model on the DTM, as they group tokens into topics (or factors). This grouping is usually done based on an assessment on which tokens co-occur with one another. The factors or topics can also yield more easily interpretable text features. Topic models have witnessed wide success in the area of information retrieval and, to this day, enjoy strong popularity in the research domains of economics and finance.

#### 2.2.3.1 Unsupervised Topic Models

**Latent Semantic Analysis**

Probabilistic topic modelling has its roots in research on dimensionality reduction methods of text features, most notably latent semantic analysis[4] (LSA) (Deerwester et al., 1990).

LSA runs a singular value decomposition of the DTM ($\boldsymbol{X}$), which might be re-weighted, for instance via tf-idf weighting. We decompose $\boldsymbol{X}$ into the matrices of its left-singular vectors $\boldsymbol{U}$ and its right-singular vectors $\boldsymbol{V}$, both having orthogonal columns. $\boldsymbol{\Sigma}$ is the diagonal matrix of singular values,

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^{\top}. \tag{2.7}$$

We can then rank the singular values in $\boldsymbol{\Sigma}$ by size and only keep the largest $K$ that still explain most of the variance in the document-term matrix, leaving us with an approximation to $\boldsymbol{X}$, which we shall label $\widehat{\boldsymbol{X}}$, such that

$$\boldsymbol{X} \approx \widehat{\boldsymbol{X}} = \boldsymbol{U}_K\boldsymbol{\Sigma}_K\boldsymbol{V}_K^{\top}. \tag{2.8}$$

In our definition of the document-term-matrix $\boldsymbol{X} \in \mathbb{R}^{D\times U}$, the left singular vectors represent the word-to-topic loadings and the right singular vectors topic-to-document loadings. By condensing the information into $K$ many factors or themes, LSA laid the conceptual foundation for the development of probabilistic topic models.

[4]Sometimes also referred to as latent semantic indexing (LSI)

However, the probabilistic assumptions underlying the LSA model do not match empirical text data characteristics. LSA's singular value decomposition of the DTM is motivated by linear algebra and assumes a joint Gaussian generative model for tokens and documents (Hofmann, 1999) – an assumption that is inadequate for discrete text data.

### Probabilistic Latent Semantic Analysis

Hofmann (1999) therefore proposed the probabilistic LSA model (pLSA), which assumes a multinomial generative process. pLSA creates topic mixtures based on a statistical latent class model and achieved sizeable performance improvements over LSA (Hofmann, 1999). In the pLSA model, the probability of token position $\tau_{d,n}$ being filled with token $v$ equals the sum of the probabilities that token $v$ has been generated by each of the $K$ different topics (or factors). Each of these probabilities themselves are calculated by two components: $\theta_{d,k}$ and $\beta_{k,v}$. $\theta_{d,k}$ is the probability that topic $k$ has been drawn for a token position in document $d$. Let $z_{d,n}$ represent the topic class for token position $\tau_{d,n}$. Thus, $p(z_{d,n} = k|\boldsymbol{\tau}_d) = \theta_{d,k}$. $\beta_{k,v}$ is the probability that token $v$ has been drawn from topic $k$. Given that topic $k$ has been allocated to token position $\tau_{d,n}$ (i.e. $z_{d,n} = k$), we can now express $p(\tau_{d,n} = v|z_{d,n} = k) = \beta_{k,v}$. The pLSA model associates an unobserved topic assignment variable $z \in \boldsymbol{Z} = \{1, ..., K\}$ with each observed token. The generative process in the pLSA model for a token position $\tau_{d,n}$ in document $\boldsymbol{\tau}_d$ to be filled with a token $v$ from the vocabulary can then be described by

$$p(\tau_{d,n} = v) = \sum_{k=1}^{K} \theta_{d,k}\beta_{k,v} = \sum_{k=1}^{K} p(z_{d,n} = k|\boldsymbol{\tau}_d)p(\tau_{d,n} = v|z_{d,n} = k). \tag{2.9}$$

Then for each document in the corpus, the joint probability model over an observed document $\boldsymbol{\tau}_d$ and a token $w$ is defined by the mixture

$$p(\boldsymbol{\tau}_d, \tau_{n,d}) = p(\boldsymbol{\tau}_d) \sum_{k=1}^{K} p(z_{d,n} = k|\boldsymbol{\tau}_d)p(\tau_{d,n} = v|z_{d,n} = k) \qquad \forall n \in N_d, \forall d \in D \tag{2.10}$$

A document and the tokens are conditionally independent given the latent topic $k$.

We can now describe LSA's generative process with two well defined steps (based on (Hofmann, 1999; Staines, 2014):

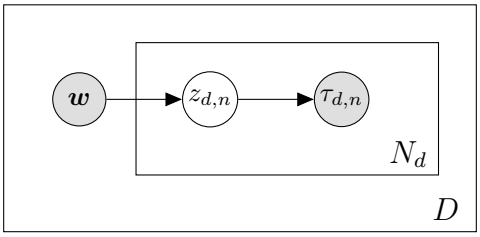
1. For each token position $\tau_{d,n}, n \in N_d, d \in D$ in each document, draw a topic assignment

$z_{d,n}$ independently from the categorical distribution $\theta_{d,k} = p(z_{d,n} = k|\boldsymbol{\tau}_d)$

2. For each token position $\tau_{d,n}$ in each document, draw the token independently from the categorical distribution $\beta_{k,v} = p(\tau_{d,n} = v|z_{d,n})$

The parameter estimation of the model can be conducted via an expectation-maximization algorithm (Dempster et al., 1977). For further technical detail on the estimation procedure, see Hofmann (1999). Figure 2.1 shows the graphical model of pLSA. Note that in the pLSA model, $\boldsymbol{\tau}$ is a dummy index for a multinomial random variable with its dimensionality matching the number of the training set documents (Blei et al., 2003). The major drawback of pLSA is that it can only learn the the topic mixtures $p(\boldsymbol{z}|\boldsymbol{\tau})$ for documents in the training set - it cannot naturally assign topic mixtures to previously unseen documents (Blei et al., 2003). Furthermore, pLSA is prone to overfitting.

Figure 2.1: Graphical model for pLSA

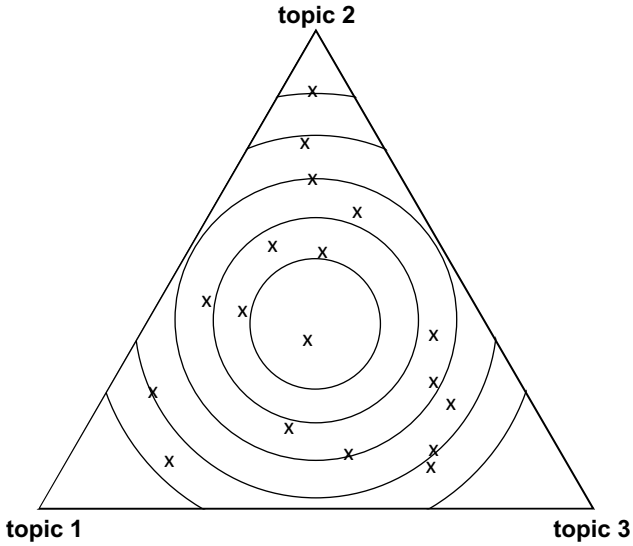



*Based on Hofmann (1999)*

### Latent Dirichlet Allocation

The limitation of pLSA gave rise to the conceptual framework of the latent Dirichlet allocation (LDA) model class (Blei et al., 2003). This concept laid the foundation for the majority of modern probabilistic topic models. As shown in Figure 2.3, LDA has a similar structure to pLSA but impose a Dirichlet prior distribution with hyperparameter $\alpha$ on the latent topic-document parameter $\boldsymbol{\theta}$ and, in its smoothed version, a Dirichlet prior distribution with hyperparameter $\eta$ on latent topic-word parameter $\boldsymbol{\beta}$. As mentioned previously, pLSA learns the topic-document distribution, $p(\boldsymbol{z}|\boldsymbol{\tau})$, only for its training documents. This leads to a learned empirical distribution of the topic mixtures ($\boldsymbol{\theta}$), which are discrete points in the topic simplex (see Figure 2.2). pLSA cannot naturally generate such topic mixtures for unseen documents, i.e. for points in the simplex that the model has not used for training. LDA places a smooth distribution on the

Figure 2.2: 3-topic simplex: comparison of pLSA and LDA




*Based on Blei et al. (2003). The x'es depict the empirical topic mixture distribution in pLSA. The contour lines depict the smoothed topic mixture distribution in LDA.*

topic-document simplex to remedy this limitation, which corresponds to contour lines in the topic simplex. Effectively, LDA uses a k-parameter latent random variable for the document-term parameter $\boldsymbol{\theta}$, whereas pLSA directly paired a document specific topic-document value to each training set document (see also (Blei et al., 2003) for more details). This allows LDA to generate topic mixtures for previously unseen documents. It also regularizes the model so that it does not tend to be prone to the same level of overfitting as pLSA (Blei et al., 2003).

The generative process of the LDA model can be described in the following steps:

1. Draw topic-document mixtures $\boldsymbol{\theta}_d$ for all $D$ documents from a Dirichlet distribution parametrised by $\alpha$: $\boldsymbol{\theta}_d|\alpha \sim \text{Dir}(\alpha)$, where $\alpha$ is a hyperparameter

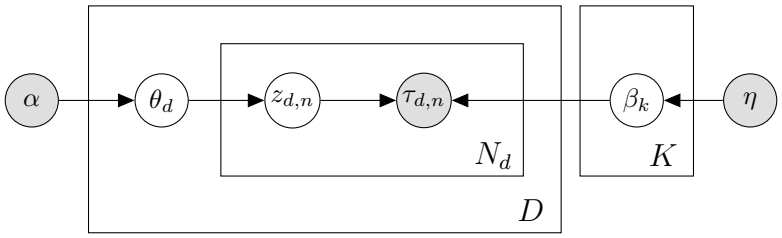
2. Draw topic distributions $\boldsymbol{\beta}_k$ for all $K$ topics from a Dirichlet distribution parametrised by $\eta$: $\boldsymbol{\beta}_k|\eta \sim \text{Dir}(\eta)$, where $\eta$ is a hyperparameter

3. For each token position $\tau_{d,n}, n \in N_d, d \in D$ in each document, draw a topic assignment $z_{d,n}$ independently from a the multinomial distribution $\theta_{d,k} \sim \text{Multi}(\boldsymbol{\theta}_d)$

4. For each token position $\tau_{d,n}$ in each document, draw the token independently from the multinomial distribution $\tau_{d,n} \sim \text{Multi}(\boldsymbol{\beta}_{z_{d,n}})$

LDA's generative process for a document $\boldsymbol{\tau}_d$ is defined by the joint probability distribution

Figure 2.3: Graphical model for LDA




*Based on Blei et al. (2003)*

over its hidden and observed variables (Blei, 2012) (and given the hyperparameters), which is

$$p(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{\mathcal{T}} | \alpha, \eta) = p(\boldsymbol{\theta}_d | \alpha) \prod_{k=1}^{K} p(\boldsymbol{\beta}_k | \eta) \left( \prod_{n=1}^{N_d} p(z_{d,n} | \boldsymbol{\theta}_d) p(\tau_{d,n} | \boldsymbol{\beta}_k, z_{d,n}) \right) \qquad \forall d \in D. \quad (2.11)$$

For parameter estimation, we then need to compute the posterior distribution (that is the conditional distribution) of the hidden variables given the observed documents,

$$p(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{z} | \boldsymbol{\mathcal{T}}, \alpha, \eta) = \frac{p(\boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}, \boldsymbol{\mathcal{T}} | \alpha, \eta)}{p(\boldsymbol{\mathcal{T}})}. \quad (2.12)$$

In the nominator of the right-hand-side in equation 2.12, we have the joint distribution from equation 2.11, which can be computed in a straightforward manner. However, the denominator (also called the evidence) in equation 2.12, $p(\boldsymbol{\mathcal{T}})$, represents the marginal probability of the observed documents. In other words, this equates to the probability of observing the corpus at hand, $\boldsymbol{\mathcal{T}}$, under any topic model configuration (Blei, 2012). Whilst marginalising out all the possible hidden variable configuration possibilities is theoretically possible, it quickly becomes computationally infeasible. We apply statistical methods for approximating the posterior. Common approaches are Gibbs-sampling, variational inference, and neural variational inference. Steyvers and Griffiths (2007) provide a comprehensive overview on Gibbs-sampling for topic models. Hoffman et al. (2013) and Blei et al. (2018) extensively cover variational inference for topic models. Srivastava and Sutton (2016), Miao et al. (2016) and Miao et al. (2017) outline the development of neural network based topic models that can be solved via neural variational inference.

In probabilistic topic models such as LDA, all documents have been generated by the same $K$ topics. However, the topic mixture is document-specific. Each topic is a specific distribution

over the vocabulary. Intuitively, these topic distributions, $\boldsymbol{\beta}$ represent the underlying semantic themes of the documents. In the end, these topic distributions over the vocabulary assess how frequently tokens co-occur with one another across documents.

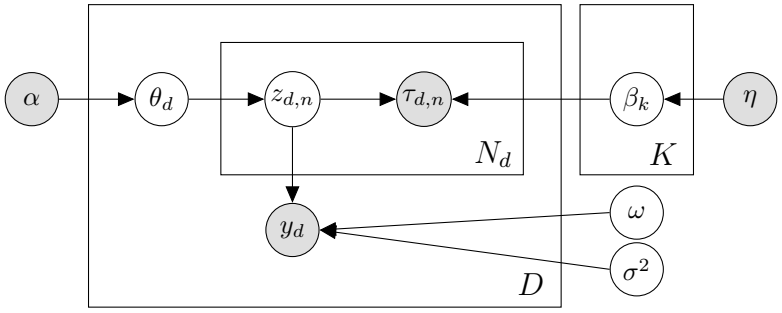
Many extension have been developed based on the unsupervised LDA concept. Particularly notable, topic models that capture correlations between topics (Blei and Lafferty, 2005), can model temporal dynamics of topics (Blei and Lafferty, 2006), and automatically determine the appropriate number of topics (Teh et al., 2005).

#### 2.2.3.2 Supervised Topic Models

The factor or topic models so far have been based on unsupervised learning algorithms. In unsupervised models, the topics are usually defined so that the maximal variance of the data is explained or the reconstruction loss of the data is minimized. However, in many applications there exists a target variable that one wants to predict based on the information in the text data. In economics and finance, examples could be to predict company default risks based on company reports or predicting market reactions to central bank statements.

Supervised topic models allow to add an additional term to the model's loss function to capture how well the extracted factors or topics predict the target variable, whilst also aiming to reduce the text data reconstruction error. In their seminal paper, Blei and McAuliffe (2008) proposed the supervised LDA (sLDA) model. Their paper demonstrated that supervised topic models should outperform unsupervised topic models in tasks such as sentiment analysis or classification tasks where additional target variable data can be leveraged to improve predictive performance.

Figure 2.4: Graphical model for sLDA




*Based on Blei and McAuliffe (2008)*

Figure 2.4 depicts the graphical model of sLDA. Based on the model of LDA, sLDA has an

additional term for the target variable $y_d$ which is document-specific. The topic assignments $\boldsymbol{z}$ not only have to generate the text data with minimal reconstruction loss, but also to predict the target variable according to some loss function metric, which is in sLDA (and many other supervised topic models) mean squared error loss for regression tasks and cross entropy loss for classification. sLDA applies a normal linear prediction model, such that $y_d|z, \omega, \sigma^2 \sim N(\omega^\top \bar{z}_d, \sigma^2)$, where $\omega$ is the mean parameter, $\sigma^2$ is the variance parameter, and $\bar{z}_d = \frac{1}{N_d}\sum_{n=1}^{N_d} z_{d,n}$. The generative process is the same as for LDA and then adds one additional step at the end:

1. Generative process for LDA
2. For each document, draw target variable $y_d$ from a normal distribution, such that $y_d|z, \omega, \sigma^2 \sim N(\omega^\top \bar{z}_d, \sigma^2)$

Over the years, the research community brought forward several extensions and improvements on supervised topic models. As an example, while Blei and McAuliffe (2008) optimize their sLDA model with respect to the joint likelihood of the document data and the response variable using variational inference (VI), MedLDA (Zhu et al., 2012) optimizes with respect to the maximum margin principle, Spectral-sLDA (Wang and Zhu, 2014) proposes a spectral decomposition algorithm, and BPsLDA (Chen et al., 2015) uses backward propagation over a deep neural network. Card et al. (2018) use neural variational inference for their SCHOLAR model which can be trained as an unsupervised, supervised, or supervised multimodal topic model. These later models all demonstrated performance improvements on supervised tasks over the original sLDA model.

## 2.3 Context-Based Token Representations and NLP Models

### 2.3.1 Word Embeddings

In this section, we cover how to obtain distributed text representations (from here on also just referred to as word embeddings) via a neural network model. We will cover the foundations of the workings of neural networks. For a thorough review of (deep) neural networks, we refer the reader to Goodfellow et al. (2016). We will also cover the neural network architecture and word embedding training for Word2Vec's CBOW and Skip-gram method (Mikolov et al., 2013a). In the subsequent section, the word embedding training logic will then be encapsulated into

the transformer architecture (Vaswani et al., 2017), which is the neural network architecture underlying virtually all modern state-of-the-art (SOTA) large language models.

#### 2.3.1.1 One-Hot Word Embeddings

The token representation methods covered so far can be classified as count-based methods. In those approaches, we have treated words (or n-grams) as atomic units. The only information we have retrieved from the raw text has been the frequency of occurrence. Hence the term bag-of-words approach that we came across previously, since the information of the order of the token occurrences had been neglected.

In those count-based approaches, every token in our vocabulary can be thought of as being simply represented by a one-hot vector, indicating the token's position in the vocabulary list. If we take the tokenized version our previously used sample sentence [I, live, in, New_York], we can represent each token as a one-hot vector such that:

$$\text{i} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \text{live} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \text{in} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad \text{New\_York} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

In the previous count-based methods, we applied different approaches that leveraged the frequency information of each token. However, we did not encode any information about semantic or syntactic similarity between words. The different vectors do not encapsulate any form of word meaning.

Apart from ignoring the word order and with it the encoded semantic and syntactic information, such one-hot word vectors also leads to computational inefficiency and therefore high computational costs on large datasets. Each one-hot word vector has dimensionality $U$ (number of unique tokens), which can easily reach 20,000-50,000 in larger corpora. However, the only information encoded is the vocabulary position.

#### 2.3.1.2 Distributional Word Embeddings

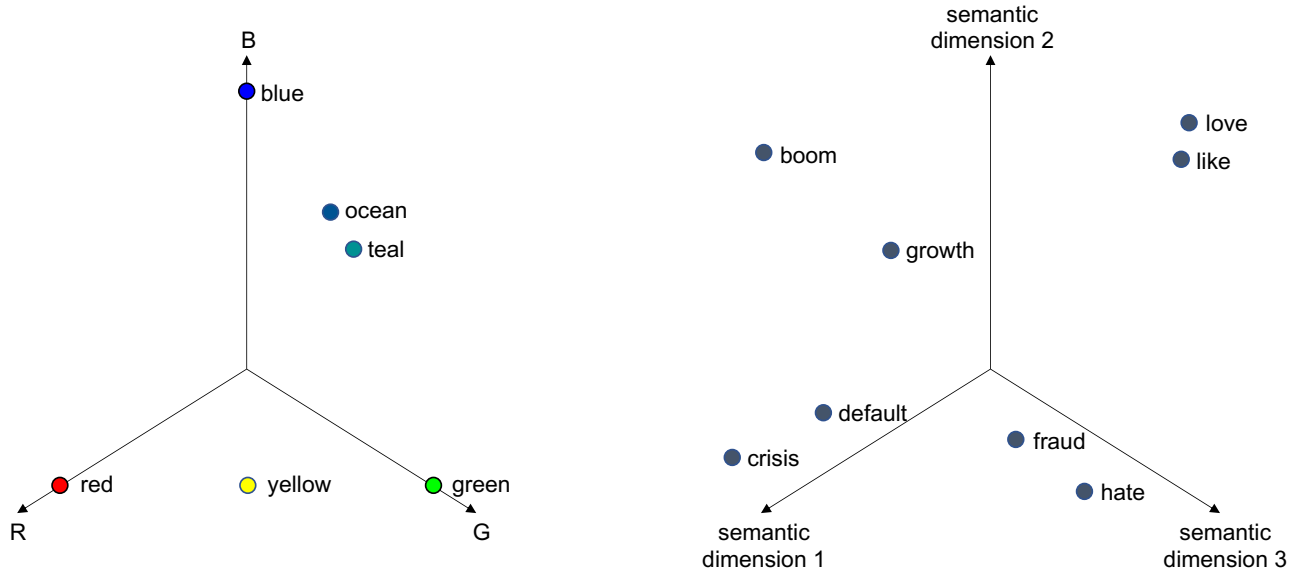
Conceptually, previous methods such as word count approaches, LSA, and topic models use the document-level as context. That is, the relative frequency of a word compared to other words is what provides the information to distinguish one document from the other. In a way, document-level models capture semantic *relatedness* between words (e.g. 'growth' and 'economy').

An alternative to this are distributed token representations (Hinton et al., 1986) that aim to leverage the word-level as context, encoding semantic and syntactic meaning of a word directly in its embedding. Such approaches attempt to capture the *similarity* between words (e.g. 'growth' and 'increase'). Linguist Firth (1957) famously summarised the core principle underlying the distributed token representations as "*you shall know a word by the company it keeps*".

An analogy to the system of colours can be helpful to better understand the concept of a multi-dimensional semantic word space. Every colour that exists, can be uniquely described according to its red, green, and blue (RGB) mix. We can therefore map every colour into a 3-dimensional space in which each axis corresponds to one of the RGB components. Colours that are close to one another in this RGB-space share similar proportions of red, blue and/or green and therefore look similar to us.

The same concept applies to our semantic word space. We map all words into an E-dimensional space. Each dimension represents some latent semantic characteristic. It is less clear how many dimensions (E) this semantic word space should have – and modern word embeddings often range from 50 to 500+ dimensions. But the logic is the same. Words that lie close to one another in the semantic word space, are semantically close to one another. Figure 2.5 illustrates the analogy between the colour and the word vector space.

Figure 2.5: Analogy between colour space and semantic space




Many different methods exist to calculate distributional word embeddings. Seminal methods were the Word2Vec approaches of continuous-bag-of-words (CBOW) and Skip-gram (Mikolov et al., 2013a) which are based on shallow neural networks, GloVe (Pennington et al., 2014) which combines global matrix factorization and local context window methods, and FastText (Bojanowski et al., 2017) which is an improvement building upon the Skip-gram approach. Modern large language models (LLMs) based on the transformer architecture (Vaswani et al., 2017) (which will be covered in section 2.3.2) such as encoder-only transformers like BERT (Devlin et al., 2018) or decoder-only transformers like GPT-1/2/3/4 (Radford et al., 2018, 2019; Brown et al., 2020; OpenAI, 2023) create word embeddings following a conceptually similar logic and achieve state-of-the-art performance in many classical NLP tasks.

The conceptual approach is to calculate the distributed word embedding by taking into account the context around a target word. This context can either be a window of words neighbouring the target word in which the exact word order is ignored, such as in CBOW and Skip-gram, or where the exact word position is encoded in the word embeddings such as in BERT.

In the following section, we outline the Word2Vec (CBOW and Skip-gram) methods before we cover the transformer architecture. The Word2Vec model is relatively simple and its general word embedding training logic to predict a missing word given its neighbouring contextual words stills finds itself being used in modern large language models that use masked language

modelling (MLM) for their self-supervised pre-training stage.

#### 2.3.1.3 Learning Word Embeddings with Neural Networks

A neural network (NN) is a parametric model that aims to approximate some unknown function $f$. This approximation is achieved through the use of a combination of, so called, neurons which are simple parametric functions. These neurons are usually arranged in layers. Hence, a basic neural network can be described as a series of functional transformations so that each neuron represents a (non-)linear function of a linear combination of the inputs, where the coefficients in the linear combination are parameters that are ultimately being trained by feeding data into the model (Bishop, 2006). The simplest neural network structure is the multilayer perceptron (MLP) consisting of one input layer of $U$ neurons represented as an input column-vector $\boldsymbol{x} = [x_1, ..., x_u, ..., x_U]^\top \in \mathbb{R}^U$, and one hidden layer with $E_0$ many neurons represented as a column-vector $\boldsymbol{h}^{(0)} = [h_1^{(0)}, ..., h_e, ..., h_{E_0}^{(0)}]^\top \in \mathbb{R}^{E_0}$. The output layer would have $C$ many neurons if we faced a classification task with $C$ many classes, for example. In case of classification, often a softmax function is then added as the last activation function. We label the output column vector $\boldsymbol{y} = [y_1, ..., y_c, ..., y_C]^\top \in \mathbb{R}^C$. In case of regression, $C = 1$. In the hidden layer of the MLP, each neuron is fully connected to all neurons of the previous layer. The net input into the hidden layer (that is the input before any activation function is applied) is

$$a_e^{(0)} = \sum_{u=1}^{U} w_{u,e}^{(0)} x_u + b_e^{(0)} \tag{2.13}$$

and each hidden unit $h_e^{(0)}$ is defined as

$$h_e^{(0)} = f^{(0)}(a_e^{(0)}) = f^{(0)}\left(\sum_{u=1}^{U} w_{u,e}^{(0)} x_u + b_e^{(0)}\right). \tag{2.14}$$

Here, $w_{u,e}^{(0)}$ are the weight parameters, and $b_e^{(0)}$ is the bias parameter of hidden unit $h_e$ in hidden layer (0) (we only have one hidden layer in this example). The activation function that is being applied to each neuron in the respective layer is depicted by $f$. Popular activation functions chosen in the literature are, for instance, the sigmoid (also called logistic) function, the tanh function, or the rectified linear unit (ReLu) function $\sigma_{ReLu}(e) = \max\{e, 0\}$. We have now (usually non-linearly) transformed the input $\boldsymbol{x} \in \mathbb{R}^U$ into the embedding space of the hidden

layer $\boldsymbol{h} \in \mathbb{R}^{E_0}$. If $U \neq E_0$ we have also changed the dimensionality of our input from $U$ to $E_0$. In the output layer, all neurons are connected to all neurons in the hidden layer, so that for each unit in the output layer, we get

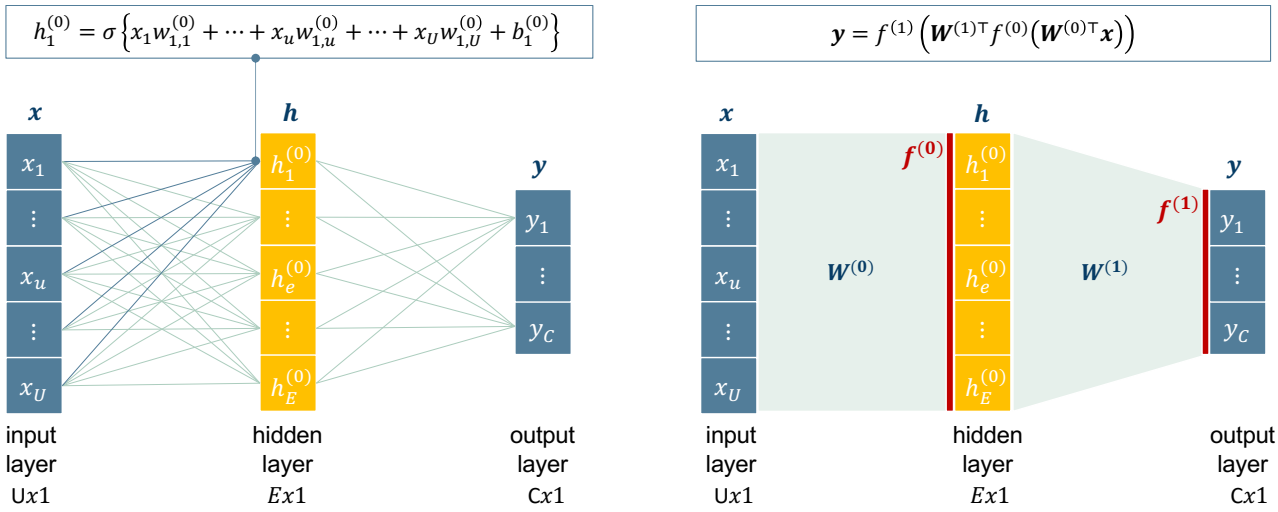
$$y_c = f^{(1)}(\ a_c^{(1)}) = f^{(1)}(\sum_{e=1}^{E} w_{e,c}^{(1)} h_e^{(0)} + b_c^{(1)}). \tag{2.15}$$

These equations fully specify the MLP. Figure 2.6 depicts the neural network structure of an exemplary feedforward neural network with one hidden layer and a sigmoid function as the activation function $f$. Such models can theoretically have arbitrarily many hidden layers $\boldsymbol{h}^{(k)}$ with $E_k$ neurons. The parameters to be trained in such a model are the weight and bias parameters across all hidden and output layers. We represent them by a matrix $\boldsymbol{W}$.

To learn those model parameters $\boldsymbol{W}$, we first need to specify an objective loss function, in which we usually compare output $y_c$ against a gold standard target value $t_c$. We could for instance use a residual sum of squares (RSS) as a loss function for a regression problem, or cross-entropy loss for classification (Bishop, 2006). We optimize our loss function by differentiating it with respect to our weight and bias parameters and backpropagating that loss gradient through the network to update the respective model parameters. For further technical details, see Goodfellow et al. (2016).

This simple example serves to explain the basic structural concept of a neural network. If we imagined $\boldsymbol{x}$ to be a one-hot vector of a word, say $I = [1, 0, 0, 0]$, from our example above, we can interpret $\boldsymbol{h}$ as the word embedding that the neural network learned to achieve the best performance, say, given a classification or reconstruction task. In the following sections, we shall cover a few NLP-specific neural network architectures that build upon this concept. Furthermore, we will look at the specific optimization tasks that the neural network will be trained on.

Figure 2.6: Architecture of a feedforward neural network

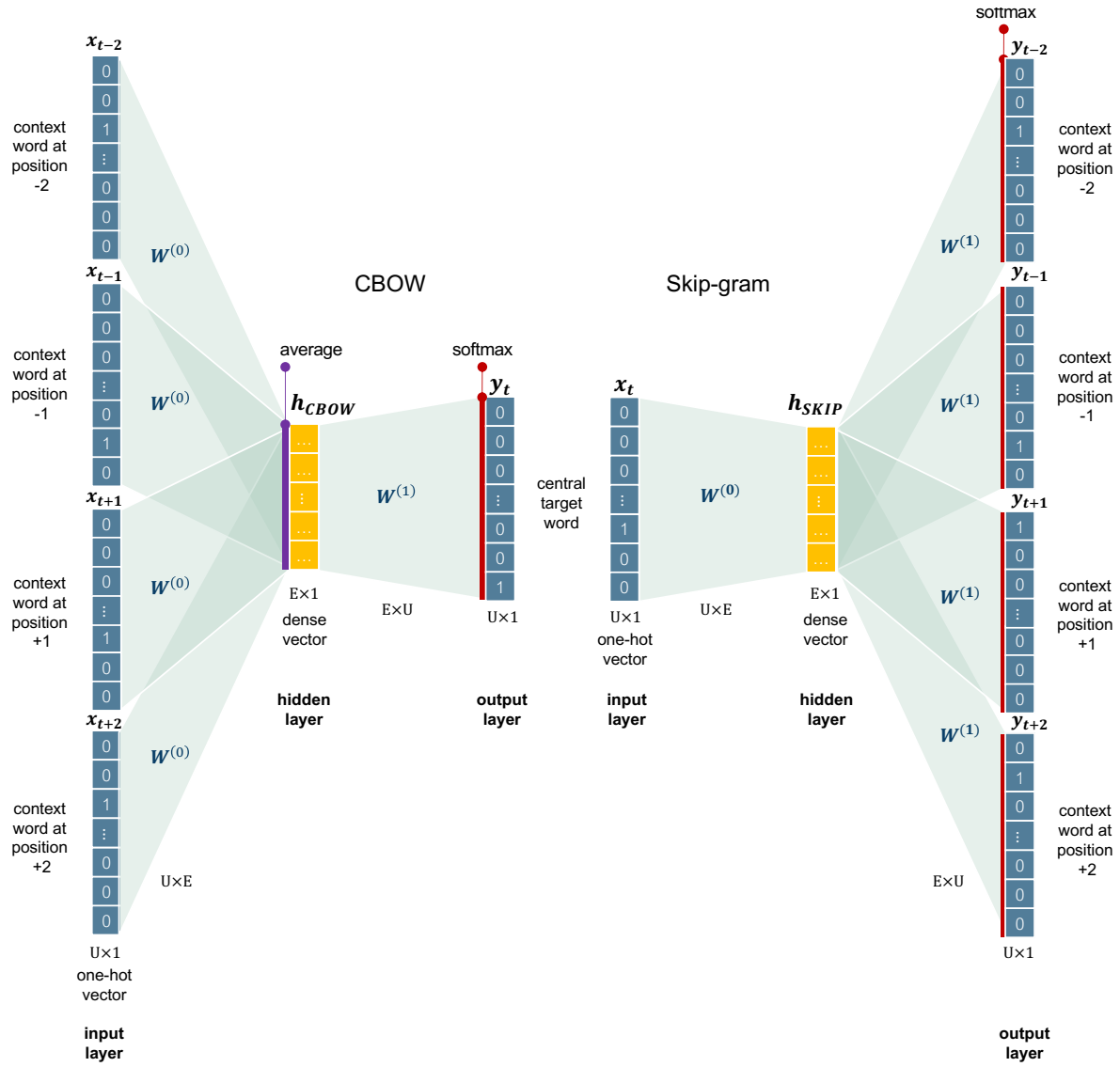



*Left-hand side depiction focuses on individual neurons. Right-hand side depiction focuses on transformation of input data.*

#### 2.3.1.4 Word2Vec: CBOW and Skip-gram

In their seminal paper, Mikolov et al. (2013a) introduced Word2Vec, a method that encapsulates two related shallow neural network models to learn word embeddings. The two approaches – continuous-bag-of-words (CBOW) and Skip-gram – differ in their training objective and in the neural network architecture. In CBOW, the model is given a set of context words and has to predict the missing (or masked out) central word of the given context window of words. Skip-gram flips the process. It is only given the central word and has to predict the context words surrounding the central target word. The overall principle is the same in both approaches: leverage contextual information to encode the semantic meaning of a word in a vector representation. Both methods are deploying a *self-supervised learning* approach. That is, by masking out or withholding parts of the data sequence in the text dataset, we create a supervised learning objective for which the ground truth label comes directly from the text itself and needs no further (manual) annotation or data-processing work. Self-supervised learning has become the cornerstone to train modern language models on vast amounts of data. Figure 2.7 outlines the neural networks architectures for both CBOW and Skip-gram.

Figure 2.7: CBOW and Skip-gram neural network structure




*Based on Mikolov et al. (2013a), with window size 5.*

## CBOW Model

The CBOW model (see Figure 2.7, left-hand side) takes as an input a set of context words within a context window of length $C$ and predicts the target word. The context window then slides over the input sequence so that each word has been the target word at some point. Table 2.2 gives an example how a sliding context window ($C = 5$) would be applied to the tokenized sample sentence: [you, shall, know, a, word, by, the, company, it, keeps].

Table 2.2: Example of applying CBOW's context window

| **Sliding context window (C=5)** | **Context words** | **Target word** |
|---|---|---|
| you, shall, know | -, -, shall, know | you |
| you, shall, know, a | -, you, know, a | shall |
| you, shall, know, a, word | you, shall, a, word | know |
| shall, know, a, word, by | shall, know, word, by | a |
| know, a, word, by, the | know, a, by, the | word |
| a, word, by, the, company | a, word, the, company | by |
| word, by, the, company, it | word, by, company, it | the |
| by, the, company, it, keeps | by, the, it keeps | company |
| the, company, it, keeps | the, company, keeps, - | it |
| company, it, keeps | company, it, -, - | keeps |

We can start understanding the workings of the CBOW model by considering its simplest version, in which the context window would include only one word. In this case, only one context word is fed into the input layer. In the input layer, the context word is encoded as a one-hot vector, which has the length of the vocabulary $U$. The hidden layers in both CBOW and Skip-gram have linear activation functions. We can define the hidden layer in CBOW as

$$\boldsymbol{h}_{CBOW} = \boldsymbol{W}^{(0)\top}\boldsymbol{x}, \tag{2.16}$$

where the one-hot input vector of the context word is defined as $\boldsymbol{x} \in \mathbb{R}^{U\times 1}$. $\boldsymbol{W}^{(0)} \in \mathbb{R}^{U\times E}$ is the matrix containing the weight parameters that transform the input layer into the hidden layer. Finally, $\boldsymbol{h}_{CBOW} \in \mathbb{R}^{E\times 1}$ is the hidden layer (and the word embedding) with dimensionality $E$. The dimension of $E$ is a hyperparameter of the model. In practice, $E$ is usually set somewhere between 50-1000 (Mikolov et al., 2013a). Each row of $\boldsymbol{W}^{(0)}$ represents the $E$-dimensional word embedding of its corresponding word in the input layer, since for each one-hot context vector $\boldsymbol{x} = [x_1, ..., x_u, ...x_U]$, $x_j = 1$ and $x'_j = 0$ for $j' \neq j$. That is, given the one-hot encoding of $\boldsymbol{x}$, the hidden layer simply copies and transposes the $j$-th row ($\boldsymbol{w}_j^{(0)}$) of the weight matrix $\boldsymbol{W}^{(0)}$, such that

$$\boldsymbol{h}_{CBOW} = \boldsymbol{W}^{(0)\top}\boldsymbol{x} = \boldsymbol{w}_j^{(0)\top}. \tag{2.17}$$

The output layer is then connected to the hidden layer via a different parameter weight matrix $\boldsymbol{W}^{(1)} \in \mathbb{R}^{E\times U}$, making the output layer again $U$-dimensional. The multiplication of the hidden

layer (word embeddings) with $\boldsymbol{W}^{(1)}$ creates a score $\boldsymbol{a} \in \mathbb{R}^{U \times 1}$ for each of the $U$ unique words that define the length of the output vector, such that

$$\boldsymbol{a} = \boldsymbol{W}^{(1)\top} \boldsymbol{h}_{CBOW}. \tag{2.18}$$

For each word, we can express the score as

$$a_u = \boldsymbol{w}_u^{(1)\top} \boldsymbol{h}_{CBOW}. \tag{2.19}$$

We can then apply for example, a softmax function over the scores, to obtain the predicted probability for each word $x_u \in U$ given the context

$$p(x_u | x_j = 1) = y_{t,u} = \frac{\exp(a_u)}{\sum_{u'=1}^{U} \exp(a_{u'})}, \tag{2.20}$$

where $y_{t,u}$ is the $u$-th entry of the output layer column representing the predicted probability that target word $t$ is the $u$-th word in the vocabulary.

Leveraging equations 2.17 and 2.19, we can rewrite equation 2.20 as

$$p(x_u | x_j = 1) = \frac{\exp(\boldsymbol{w}_u^{(1)\top} \boldsymbol{w}_j^{(0)\top})}{\sum_{u'=1}^{U} \exp(\boldsymbol{w}_{u'}^{(1)\top} \boldsymbol{w}_j^{(0)\top})}. \tag{2.21}$$

From equation 2.21 we can now see that, conceptually, in the nominator inside the exponential function, we compute a dot product similarity between the word vector from the output layer weight matrix $\boldsymbol{W}^{(1)}$ for word $u$ and the word vector from the input layer weight matrix $\boldsymbol{W}^{(0)}$ for the context word $j$. We then scale or 'normalise' that value by applying the value from the denominator. Here inside exponential function, we calculate the sum of all dot product similarities between all word vectors $u \in U$ from the output layer weight matrix, with the word vector from the input layer weight matrix for word $j$.

So far, we only considered the special case of having one context word. The logic works almost identically when we consider $C$ context words in the input layer. The hidden layer is

now computed by taking the average over the input context vectors,

$$h_{CBOW} = W^{(0)\top}\frac{1}{C}(\boldsymbol{x}_1 + ... + \boldsymbol{x}_c + ... + \boldsymbol{x}_C) \tag{2.22}$$

$$= \frac{1}{C}(\boldsymbol{w}_{x_1}^{(0)\top} + ... + \boldsymbol{w}_{x_c}^{(0)\top} + ... + \boldsymbol{w}_{x_C}^{(0)\top}). \tag{2.23}$$

From thereon, everything is the same as in the one word context case. We can write the predicted probabilities as

$$p(x_u|x_{c\in C}) = \frac{\exp(\boldsymbol{w}_u^{(1)\top}\boldsymbol{w}_{\bar{j}}^{(0)\top})}{\sum_{u'=1}^{U}\exp(\boldsymbol{w}_{u'}^{(1)\top}\boldsymbol{w}_{\bar{j}}^{(0)\top})}, \tag{2.24}$$

where we denote $x_{c\in C}$ as representing all the input words in the context window, and $w_{\bar{j}}^{(0)}$ depicts the average over the input context word embeddings.

As we can see from this approach, the context around a word is used to compute the distributed word embeddings and to infer the missing word in the centre of the context window. However, the actual order of the context words does not affect the prediction. The CBOW model averages over the context vectors when information is passed from the input layer to the hidden layer. Hence, the name: continuous-bag-of-words model.

### Skip-gram Model

The Skip-gram model (see Figure 2.7, right-hand side) reverses the CBOW process: we have one target word as the input for which we predict the surrounding context words. Since we have one input word, the hidden layer definition is the same as for the CBOW model with one context word. The target word enters the input layer as the one-hot encoded vector. The hidden layer of the Skip-gram model is

$$\boldsymbol{h}_{SKIP} = \boldsymbol{W}^{(0)\top}\boldsymbol{x}_t = \boldsymbol{w}_j^{(0)\top}. \tag{2.25}$$

At the output layer, we are now predicting $C$ context words instead of one target word. For each projection from the hidden layer to the output layer, the same weight matrix $\boldsymbol{W}^{(1)}$ is used in the Skip-gram model, so that for each of the $C$ context word predictions, we get a score value for each of the $U$ entries in the output layer corresponding to the unique words in the

vocabulary. We simply get $C$ score vectors of

$$\boldsymbol{a}_c = \boldsymbol{W}^{(1)\top} \boldsymbol{h}_{SKIP}. \tag{2.26}$$

Similarly, if we now apply a softmax function, we get as our predicted probabilities

$$p(x_{c,u}|x_{t,j} = 1) = y_{c,u} = \frac{\exp(a_{c,u})}{\sum_{u'=1}^{U} \exp(a_{c,u'})}, \tag{2.27}$$

where $a_{c,u}$ represents the score for the $u$-th entry for output layer panel $c$. In the same logic as for CBOW, it follows that

$$p(x_{c,u}|x_{t,j} = 1) = y_{c,u} = \frac{\exp(\boldsymbol{w}_{c,u}^{(1)\top} \boldsymbol{w}_j^{(0)\top})}{\sum_{u'=1}^{U} \exp(\boldsymbol{w}_{u'}^{(1)\top} \boldsymbol{w}_j^{(0)\top})}. \tag{2.28}$$

Here, $y_{c,u}$ represents the predicted probabilities that, given the input target word, the context word for output panel $c$ is the $u$-th word in the vocabulary. Note that, to arrive at the notation in the denominator, all output layer panels share the same weight matrix $\boldsymbol{W}^{(1)}$ and hence $a_{c,u} = a_u = \boldsymbol{w}_u^{(1)\top} \boldsymbol{h}_{SKIP}, \forall c \in C$. Again, we can see that the actual word order within the context window has no impact on the predictions.

Both CBOW and Skip-gram are optimized using stochastic gradient descent. According to Mikolov et al. (2013a), CBOW has been observed to train faster and therefore might be preferable to train on larger datasets. According to the authors, skip-gram can yield more semantically and syntactically meaningful embeddings on smaller datasets or on datasets where the vocabulary contains a lot of rare words. Word2Vec and related methods such as GloVe paved the way for language model that are trained on distributed text representations. Word embeddings, such as Word2Vec, learn semantically meaningful word embeddings based on the words in their context window. However, the relevance weight that is given to each context word is the same, and the context is limited on the window length. Furthermore, such approaches cannot address word polysemy – the meaning of 'bank' is quite different depending on whether it is preceded by 'river' or 'commercial'. Yet, in models such as Word2Vec, there would only exists one word embedding for 'bank'. In the next section, we cover the deep learning architecture called transformer (Vaswani et al., 2017) that is the foundational model for virtually all modern language models and which enabled vast breakthroughs in SOTA-performances in classical NLP

tasks since its release in 2017.

### 2.3.2 Transformers

#### 2.3.2.1 A Brief History of Transformers

Since Word2Vec, many modern language models have been proposed to create more nuanced contextual word embeddings. A breakthrough moment for the field of NLP occurred when Vaswani et al. (2017) introduced the transformer model – a novel neural network architecture that could both address those shortcomings mentioned in the previous section with regards to capturing context whilst also achieving vast computational speedups through parallelising computational steps. In its core, the transformer model is a deep encoder-decoder neural network that leverages attention layers in various parts of its architecture. The key elements of the transformer architecture will be described in the following sections. However, it is beyond the scope of this thesis to cover the transformer model in its minute detail. Additional resources are Rush et al. (2018) who created an annotated version of the original paper, Alammar (2018) who covers and illustrates the key concepts of the transformer model, and Phuong and Hutter (2022) who provide an extensive formal description of the model's algorithmic components.

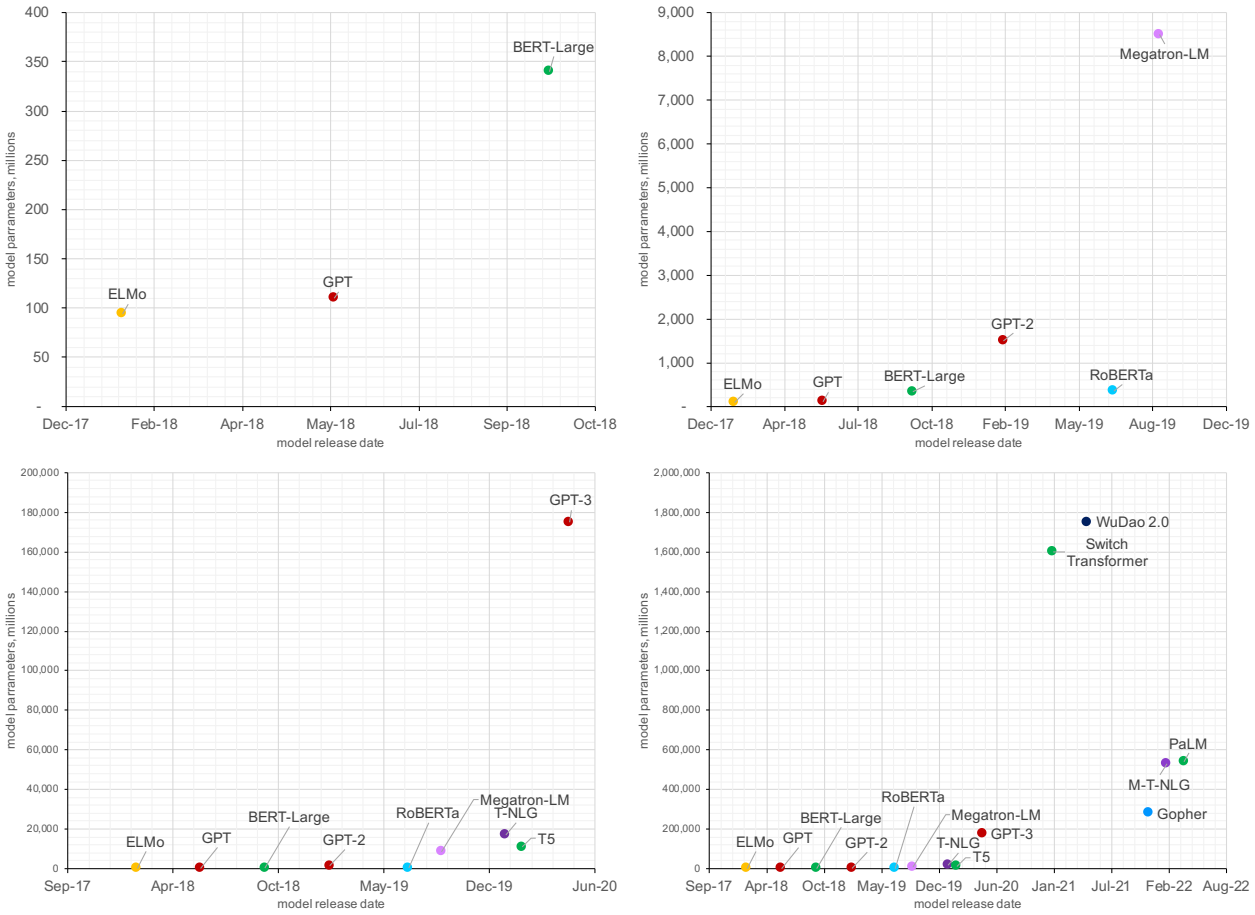
At present, virtually all modern large language models (LLMs) are based on (parts of) the transformer architecture. The transformer architecture – combined with advances in computational hardware design, particularly with respect to graphical processing units (GPUs) and tensor processing units (TPUs) – allows to build models with hundred millions to hundred billions of parameters and to train them on enormous datasets that can include the entirety of Wikipedia and most of mankind's books that are accessible in digital format. One of the key findings since the publication of the transformers paper is that so far, the marginal rate of performance return from increasing model and training set sizes still seems to be far away from zero. Table 2.3 and Figure 2.8 show the seemingly exponential growth in model parameter size between 2018-23. BERT-Large (Devlin et al., 2018) by Google achieved SOTA performance in many NLP tasks at its time of publication and has over 345 million trainable model parameters. It was considered to have reached a new stratosphere in terms of model size at its release time. About only one year later, Megatron-LM (Shoeybi et al., 2019) by Microsoft, featured 8.3 billion parameters, which is 25 times BERT-Large's size. About another half a year later,

in mid-2020, OpenAI dwarfed the NLP landscape by releasing GPT-3 (Brown et al., 2020), which has 175 billion parameters – about 500 times the size of BERT-Large. By the end of 2022, both Google (Fedus et al., 2022) and the Beijing Academy of Artificial Intelligence (BAAI) (BAAI, 2018) have released transformer-based language models with over 1.5 trillion parameters – around 10 times bigger than GPT-3 or over 5,000 times bigger than BERT-Large. Finally, as this thesis is being finished in March 2023, OpenAI has released GPT-4 (OpenAI, 2023). The official model architecture has not been released yet. However, as the ratio of model size to training time has been quite linear over past models, researchers have inferred that GPT-4 might be as large as up to 100 trillion model parameters or a combination of several multi-hundred-billion parameter models. Training those models usually costs several million US dollars so that the development of such models is by and large exclusively in the hand of large and private technology companies. Thanks to transfer learning capabilities of these models (which will be covered in this section), researchers can however download some those pre-trained LLMs that have been made publically available. Such models can then be fine-tuned on domain specific tasks at relatively small computational costs. Also, it should be noted that through continuing innovation particularly in algorithmic design, pre-training costs for models such as BERT have been reduced to around [5]$20-400. But even with the latest algorithmic computation innovations, at the time of writing this thesis, models such as GPT-3 will cost over [6]$500,000 purely in computational resources to be trained from scratch. Larger models will cost orders of magnitude more. These costs do not factor in development costs including expenses for data acquisition and preparation. For example, training and development costs for GPT-4 have been stated to lie north of [7]$100 million. In conclusion, whilst the development of such foundational language models seems – currently – to be beyond reach for academia, public access is available to some of these pre-trained models, which allows researchers to apply them to answer new research questions (often about those LLMs themselves). It is also not all about model size. For example, OpenAI achieved substantial performance boosts in dialogue tasks with their model chatGPT (OpenAI, 2022), which is a modified and much smaller version of GPT-3. OpenAI strongly leveraged human curation and annotation of datasets as well as alternative training strategies such as reinforcement learning from human feedback (RLHF) techniques to boost

[5] www.mosaicml.com/blog/mosaicbert
[6] www.mosaicml.com/blog/gpt-3-quality-for-500k
[7] www.wired.com/story/openai-ceo-sam-altman-the-age-of-giant-ai-models-is-already-over/

their model performance (OpenAI, 2022).

Figure 2.8: Model parameter sizes (in millions) of selected LLMs, 2018-23




*Model list makes no claim to completeness. Top-left panel until 2019. Top-right panel until 2020. Bottom-left panel until 2021. Bottom-right panel until 2023. Values from respective model publications and press releases.*

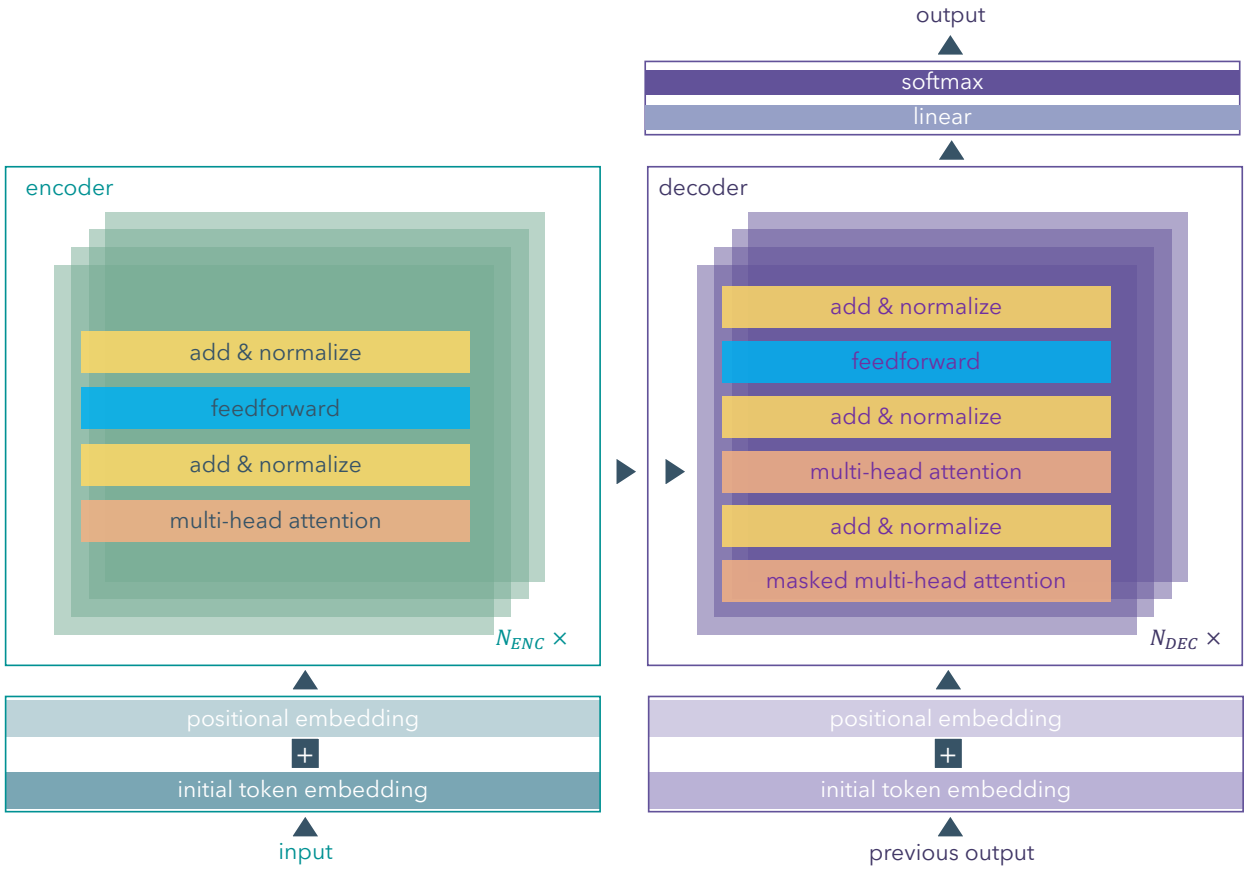
Table 2.3: Model parameter sizes (in millions) of selected LLMs, 2018-23

| **Developer** | **Model name** | **Release date** | **Parameters, in mn** |
|---|---|---|---|
| AI2 | ELMo | Feb-18 | 94 |
| OpenAI | GPT | Jun-18 | 110 |
| Google | BERT-Large | Oct-18 | 340 |
| Meta AI | RoBERTa-Large | Jul-19 | 354 |
| OpenAI | GPT-2 | Feb-19 | 1,500 |
| Nvidia | Megatron LM | Sep-19 | 8,500 |
| Google | T5 | Feb-20 | 11,000 |
| Microsoft | T-NLG | Jan-20 | 17,000 |
| OpenAI | GPT-3 | May-20 | 175,000 |
| DeepMind | Gopher | Dec-21 | 280,000 |
| Microsoft/Nvidia | Megatron-Turing NLG | Feb-22 | 530,000 |
| Google | PaLM | Apr-22 | 540,000 |
| Google | Switch Transformer | Jan-21 | 1,600,000 |
| BAAI | WuDao 2.0 | May-21 | 1,750,000 |

#### 2.3.2.2 Transformer Architecture

This section focuses on providing a conceptual overview of the general workings and the key components of the transforer model architecture. Overall, a transformer is an encoder-decoder model that takes as input a sequence of symbol representations (e.g. the tokenized sentence [This, is, a, transformer, model]), encodes the sequences into a dense vector representation (the word embedding), and subsequently takes the encoder input to decode it back into symbolic text form (classically, for example for machine translation or next word prediction tasks) in an auto-regressive fashion, one word at a time. An outline of the model architecture is shown in Figure 2.9.

Figure 2.9: Transformer architecture outline




*Based on Vaswani et al. (2017).*

**Input Embeddings**

Transformer models are usually designed to process sequences of text at a time. A usual sequence length is around 500 tokens. Particularly, in the economics and financial domain, however, this might appear limiting. Documents such a SEC filings, analyst reports, or central

bank announcements can easily exceed this word limit by orders of magnitude. The limitation on sequence length is mostly driven by the attention mechanism (that we will cover later in the section), which has $O(N^2)$ time and memory computational complexity in the standard transformer model, $N$ being the length of the input sequence. Longer-sequence models have been suggested, such as BigBird (Zaheer et al., 2020) or Longformer (Beltagy et al., 2020), which replace the full attention matrix with a sparser one. Successful application of such model designs in the economics and finance domain is yet to be demonstrated.

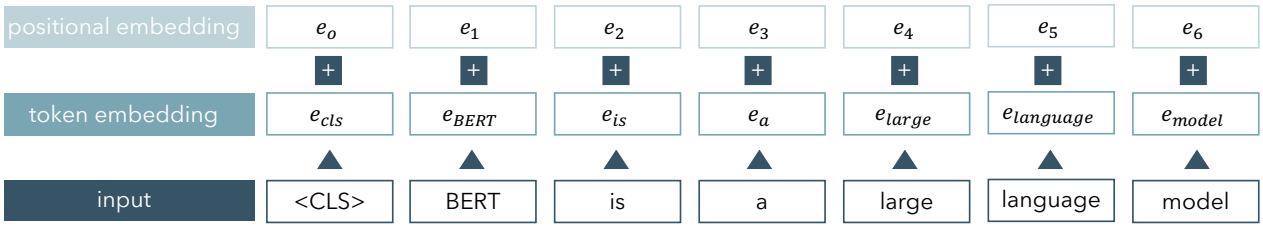
As Figure 2.9 shows, raw text inputs are first transformed into already pre-trained word embeddings. Modern LLMs oftentimes use WordPiece or BPE embeddings. However, in theory we could also use, for example, Word2Vec embeddings as a starting point. Each token $\tau$ in document $\boldsymbol{\tau} = [\tau_1, ..., \tau_{N_d}]$ (also called a sequence in many modern LLM papers) is now being represented by a distributed text representation, the initial token embedding $\boldsymbol{e}_{\tau,te} \in \mathbb{R}^{1\times d_e}$. The embedding dimensionality $d_e$ is a hyperparameter. For example, the original transformer paper Vaswani et al. (2017) use 512 dimensions to represent each token, BERT-small uses 768, BERT-large 1024 (Devlin et al., 2018). The transformer architecture's attention mechanism has no inherent method to leverage information about the location of a token in the sequence or about the distance between tokens. Therefore, a positional embedding is created for each token. These positional embeddings $\boldsymbol{e}_{\tau,pos} \in \mathbb{R}^{1\times d_e}$ can either be hard-coded or learned from the data itself.

The ultimate input embedding $\boldsymbol{e}_\tau \in \mathbb{R}^{1\times d_e}$ that is being fed into the transformer model, sums together the initial token embedding and the positional embedding for the respective token $\tau$ such that,

$$\boldsymbol{e}_\tau = \boldsymbol{e}_{\tau,te} + \boldsymbol{e}_{\tau,pos}. \tag{2.29}$$

For some transformer models, special tokens are added to the sequences. One example would be a special token that learns the overall embedding for the sequence and which can be used in downstream tasks such as sequence classification. Figure 2.10 summarises the conceptual steps from raw inputs to model input embeddings, where <CLS> symbolises the special token for the sequence classification. The entire token sequence $\boldsymbol{\tau}$ of length $N_d$ is then represented by matrix $\boldsymbol{E} \in \mathbb{R}^{N_d\times d_e}$. That is, $\boldsymbol{E}$ contains the embeddings for all tokens in the sequence.

Figure 2.10: Composition of input embeddings for transformer models




*Encoder models such as BERT add special tokens to the input sequences*

### Attention

Before describing the subsequent model layers, it is important to introduce the concept of *attention* – a key component that makes the transformer model so powerful. Intuitively, attention in transformer models represents the dependence or informativeness between tokens. That is, if we masked the token 'language' out of the sentence in Figure 2.10, how much informational value does each other word in the sequence posses to predict correctly that 'language' is indeed the word which should fill the blank in 'BERT is a large __ model'. If we as a human are presented with such a cloze question, our eyes would most likely quickly focus on the tokens 'BERT' and 'model'. Similarly, even if we replaced 'large' with 'outstanding' or 'underwhelming' (depending on where you stand on this debate), a human's prediction for the cloze answer would most likely be unaffected. However, if we replaced 'BERT' with 'DSGE'[8], the most probable prediction value for the cloze might now be 'macroeconomic'. Attention scores are high between words that are informative about one another. Oftentimes, this also means they share some form of semantic similarity or relationship with one another, i.e. what 'BERT' is for 'language', is 'DSGE' for 'macroeconomic'. In transformer models, this informativeness (i.e. attention) score is measured as the scaled dot-product between the embeddings of the respective tokens. Other forms of attention scores such as additive attention exist, but the dot-product attention score has become the dominant measure, partly also because it can be very efficiently computed leveraging highly optimized matrix multiplication code (Vaswani et al., 2017).

The implementation of the attention score is then done as follows, using the approach described in Vaswani et al. (2017). The matrix of input embeddings $\boldsymbol{E} \in \mathbb{R}^{N_d \times d_e}$ are linearly

[8]Dynamic stochastic general equilibrium. An economic model widely used in modern macroeconomic theory.

transformed into a *query matrix* $\boldsymbol{Q}$, a *key matrix* $\boldsymbol{K}$, and a *value matrix* $\boldsymbol{V}$ such that

$$\boldsymbol{Q} = \boldsymbol{E}\boldsymbol{W}_Q \quad , \boldsymbol{W}_Q \in \mathbb{R}^{d_e \times d_a} \tag{2.30}$$

$$\boldsymbol{K} = \boldsymbol{E}\boldsymbol{W}_K \quad , \boldsymbol{W}_K \in \mathbb{R}^{d_e \times d_a} \tag{2.31}$$

$$\boldsymbol{V} = \boldsymbol{E}\boldsymbol{W}_V \quad , \boldsymbol{W}_V \in \mathbb{R}^{d_e \times d_a} \tag{2.32}$$

where $\boldsymbol{W}_Q$, $\boldsymbol{W}_K$, $\boldsymbol{W}_V$ are trainable model parameters. For computational efficiency, $d_a$ is usually chosen to be smaller than $d_e$. We can think of it, as the token currently being predicted as being a vector from the query matrix. The context tokens for the prediction are represented by the key vectors that are contained in key matrix $\boldsymbol{K}$. We can think of the key and the value matrix as being the same. We can now compute the dot-product between query token vector $\boldsymbol{q} \in \mathbb{R}^{1 \times d_a}$ and each of the context token vectors $\boldsymbol{k} \in \mathbb{R}^{1 \times d_a}$ as $\boldsymbol{q}\boldsymbol{k}^\top$. The dot-product can now be interpreted as the level of informativeness of the value token to predict the query token (Phuong and Hutter, 2022). As each token will at some point be the query token, we can now efficiently calculate the dot-product between any two tokens (including a token with itself) by matrix multiplying $\boldsymbol{Q}$ and $\boldsymbol{K}^\top$. For numerical stability reasons, the resulting matrix $\boldsymbol{Q}\boldsymbol{K}^\top \in \mathbb{R}^{N_d \times N_d}$ is scaled by $\sqrt{d_a}$ before a softmax function is applied on each vector slice in the matrix. Finally, this square matrix is being multiplied with the value matrix $\boldsymbol{V}$ to weight each context token according to their 'informativeness value' to predict the respective query tokens.

$$\boldsymbol{Z} = \text{attention}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d_k}}\right)\boldsymbol{V}, \tag{2.33}$$

Here, $\boldsymbol{Z} \in \mathbb{R}^{N_d \times d_a}$ represents the token embeddings after they have been processed through the attention function (also called attention head).

**Encoder**

Usually, several encoder blocks are stacked on top of one another. That is, the first one receives the input embeddings, and the subsequent encoder blocks take as their input the output of the previous encoder block. Vaswani et al. (2017) use $N_{ENC} = 6$, BERT-small uses 12 and BERT-large uses 24 (Devlin et al., 2018). The following describes the inner layers (also called sub-layers) within a single encoder block.

The first layer of the encoder consists of a multi-head self-attention layer. That is, $N_a$

attention functions $\boldsymbol{Z}_n \in \mathbb{R}^{d_a}$ are being fed with the same input embedding $\boldsymbol{E}$ in parallel, where $N_a$ is a model hyperparameter, such that

$$\boldsymbol{Z}_n = \text{attention}(\boldsymbol{Q}_E, \boldsymbol{K}_E, \boldsymbol{V}_E) \quad , \forall \boldsymbol{Z}_n \in \mathbb{R}^{N_d \times d_a}, n = 1, ..., N_a. \tag{2.34}$$

Here, query, key, and value matrices all come from the same input sequence embedding $\boldsymbol{E}$. This is often referred to as *self-attention*. Each attention function has a different initialization in its parameter weights and therefore learns a different latent representation of the input embedding. In the original transformer paper, Vaswani et al. (2017) deploy $N_a = 8$ parallel attention functions, each with dimension $d_a = \frac{d_e}{N_a}$, which leads to the same computational complexity than one attention layer with full embedding dimensionality $d_e$. BERT-small uses 12 such attention heads per encoder block, BERT-large uses 24 (Devlin et al., 2018). Each of these $N_a$ attention functions is called an attention head. Hence, the term *multi-head attention*. The attention head outputs are then concatenated and linearly projected such that the output of the multi-head attention layer is

$$\boldsymbol{Z}_{MH} = \text{concatenate}(\boldsymbol{Z}_1, ..., \boldsymbol{Z}_{N_a})\boldsymbol{W}_A \quad , \boldsymbol{Z}_{MH} \in \mathbb{R}^{N_d \times d_e}, \tag{2.35}$$

i.e. the embedding dimension after the attention layer is again the size of the original input embedding layer. $\boldsymbol{W}_A \in \mathbb{R}^{N_a d_a \times d_e}$ is a matrix of trainable parameters.

The outputs of the multi-head attention layer $\boldsymbol{Z}_{MH}$ are then added with a residual connection (He et al., 2016) and normalized, such that

$$\boldsymbol{Z}_{AN} = \text{layernorm}(\boldsymbol{E} + \boldsymbol{Z}_{MH}) \quad \boldsymbol{Z}_{AN} \in \mathbb{R}^{N_d \times d_e}, \tag{2.36}$$

where $\boldsymbol{Z}_{AN}$ symbolises the output after the 'add & normalize' layer. Finally, the output is fed through a fully connected feedforward layer. And again the residual connection is applied and the output is being normalized

$$\boldsymbol{Z}_{ENC} = \text{layernorm}(\text{feedforward}(\boldsymbol{Z}_{AN}) + \boldsymbol{Z}_{AN}), \quad \boldsymbol{Z}_{ENC} \in \mathbb{R}^{N_d \times d_e} \tag{2.37}$$

where $\boldsymbol{Z}_{ENC}$ represents the output layer of the encoder containing the token embeddings for

each of the $N_d$ tokens in the sequence. $\boldsymbol{Z}_{ENC}$ then becomes the input for the next encoder blocker.

Some transformer models are encoder-only transformers. For example BERT (Devlin et al., 2018) or RoBERTa (Liu et al., 2019). These models usually use masked language modelling (MLM) as a training strategy[9]. That is, in the input sequence, some tokens are masked out at random and the model needs to predict the missing word. In such as case, the output layer has the dimensionality of the vocabulary and a softmax function is applied. The prediction performance can then be evaluated, for example, with respect to cross-entropy loss - similar to the process in Word2Vec. The prediction error is then backward propagated through the neutral network using Adam (Kingma and Ba, 2015).

### Decoder

The decoder of a transformer has a very similar structure to the encoder. It is also usually composed of a stack of several decoder layers $N_{DEC}$. Often the number of encoder and decoder blocks are identical though this is a pure design choice. The decoder takes the first token of the sequence and then, in an autoregressive manner, predicts then next token of the sequence. The decoder layer has all the encoder layers, but between them is an attention layer that helps the decoder to focus on relevant tokens of the input sentence. In this attention layer, the query matrix comes from the previous' decoders output, whereas key and value matrix come from the last output layer of the last encoder block. Furthermore, the decoder attention matrix has some restrictions, so that it cannot get the information of the words ahead in the input sequence that it is supposed to predict. In the decoder attention matrix, cells pertaining to subsequent positions in the sequence are masked out (or put to -inf). Some transformer models are decoder-only transformers and therefore also called autoregressive models. Models of the GPT family, for example, are decoder-only transformers.

Training loss and model parameter optimization work similarly to the encoder, only that instead of training on masked language modelling, the transformer trains on next token prediction. We can see from this, that the main differences between BERT an GPT lie in the attention masking and the training strategy.

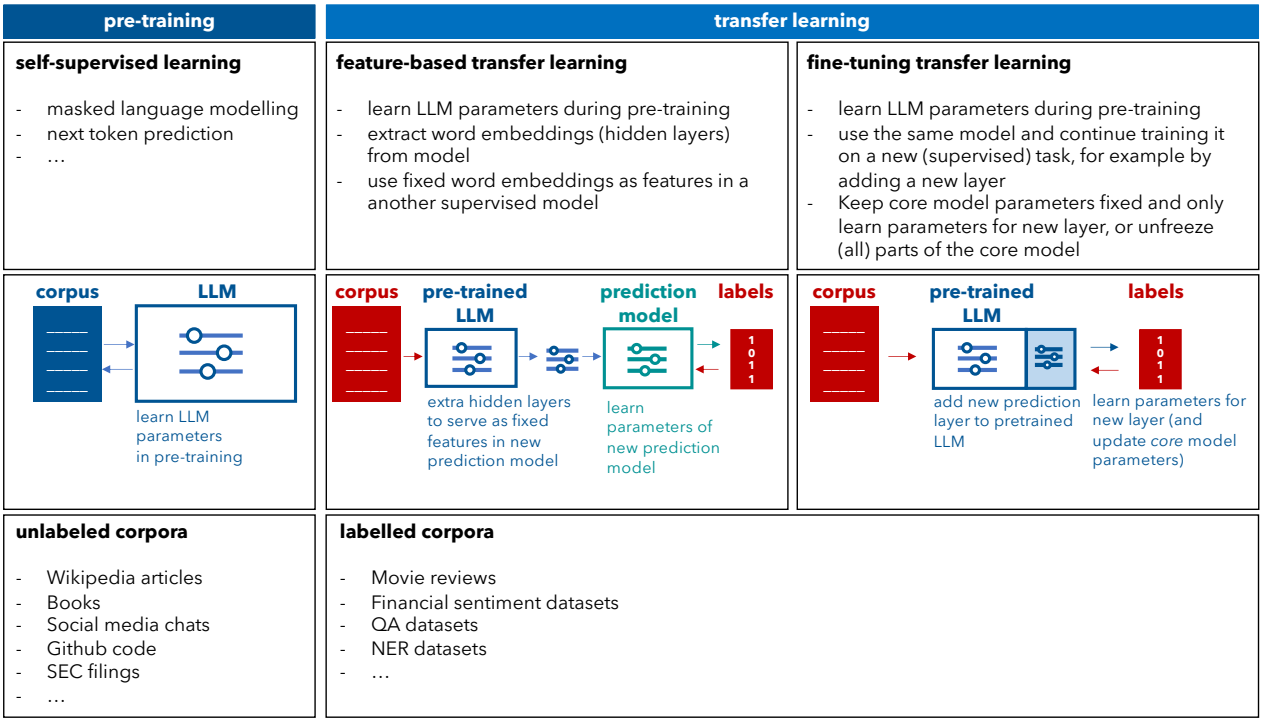
[9]The original BERT training strategy also included next sentence prediction. However, subsequent model variations dropped this and purely used MLM as the pre-training strategy

The original transformer paper by Vaswani et al. (2017) used both encoder and decoder and trained on language translation. That is, encode an English sentence, and use the decoder to reconstruct the sentence in e.g. German via next token prediction.

### 2.3.3 Pre-Training and Transfer Learning

This section briefly covers some key concepts and innovations in training such large language models. A key factor allowing single LLMs to achieve SOTA performance in a wide variety of NLP tasks, is that these models leverage the power of transfer learning. We can generally break the training procedure for LLMs into two steps - pre-training and transfer learning - as depicted in Figure 2.11.

Figure 2.11: Schematic outline of pre-training and transfer learning for LLMs




The first step is the self-supervised pre-training part, as described in the previous section. Common training strategies are masked language modelling (MLM) as in BERT or next token prediction (NTP) as in the GPT models. Pre-training usually requires vast amounts of data, substantial hardware infrastructure and long training times, which can make the pre-training part quite resource expensive. Oftentimes training costs are in the millions of pounds[10]. However,

[10]It is to be added that continuous research effort focuses on the creation of more data-efficient model constructions that can be pre-trained and fine-tuned at lower costs.

once a model is pre-trained (often called a foundational model), it can be re-adjusted for many other downstream tasks at quite low costs (affordable to an individual researcher) leveraging the concept of transfer learning.

Two main transfer learning strategies exist, as depicted in Figure 2.11. The first one is feature-based transfer learning. Here, the researcher feeds the new downstream text corpus to the pre-trained model and simply extracts the word embedding representations (i.e. hidden layers), keeping the model parameters fixed. The extracted word embeddings can then be used as features that feed into a new prediction model, which could be any supervised learning model even as simple as a linear or logistic regression. In feature-based learning, only the parameters of the new prediction model are learned, based on the downstream task dataset.

The other option, which often proves to be quite powerful, is to simply add a new output layer to the pre-trained LLM. One can then either keep the *core* model parameters fixed and only learn the parameters of the new layer, or one can unfreeze (all) parts of the core model as well and update them for the new downstream task. This parameter updating procedure is called *fine-tuning*. Fine-tuning can allow the LLM to adapt with relatively few training examples (ie. hundreds or thousands instead of million or billions during pre-training) to the new downstream task and achieve very competitive (often SOTA) prediction results. This approach is therefore particularly interesting for applications of NLP in the domain of economics and finance, in which datasets are usually relatively small compared to classic NLP datasets.

A final word on training efficiency across different transfer model architectures. Encoder-decoder transformers and encoder-only transformers have achieved SOTA performances in many NLP tasks (Chowdhery et al., 2022). Raffel et al. (2020) shows that at the same training costs, an encoder-decoder model generally outperforms a decoder-only model. However, encoder-decoder and encoder-only models tend to require much more data to achieve satisfactory fine-tuning results compared to decoder-only models. Brown et al. (2020) demonstrated with GPT-3 that very large decoder-only models can be remarkably performant few-shot learners. Few-shot learning means that the model adjust to a new downstream tasks with very few (often 5-10) training points during fine-tuning. Those results are particularly interesting for researchers in economics and finance, as it might allow to integrate LLMs into structual economics models and financial modelling pipelines and to achieve strong prediction results with relatively small

training data available. FinBERT (Araci, 2019) and FLANG-BERT (Shah et al., 2022) are fine-tuned and even further pre-trained LLMs on domain-specific financial datasets. They showing additional performance improvements on tasks such as financial sentiment classification compared to standard LLMs. The deeper integration of LLMs into existing economic and financial modelling pipelines is certainly a promising area for additional future research.

## 2.4 Multimodal Models

A crucial aspect for many research applications using observational data in economics and finance, is to adequately capture (or control for) contextual information (or covariates) about the environment in which a prediction about a certain target variable is made. If we fail to capture relevant contextual information, we can end up with a potentially substantial omitted variable bias in our model fitting - and subsequently with decreased prediction performance, as relevant information is ignored. This applies to text data as much as to classic numeric data.

A case in point for NLP in economics and finance: assume a central bank announced that it expected the inflation rate to increase by one percentage point by the end of the quarter. Is this positive, neutral, or negative news for the financial markets? The answer is: it depends. It depends on the economic environment in which this statement was delivered to the public. If we happened to be in a deflationary economy, news about an uptick in inflation might be a positive economic signal to the markets. However, if the inflation level was already much above central bank target, a further increase might be a negative market signal. And it's not only the macroeconomic conditions that needed to be captured. We would also need to understand the contemporary market expectations, for example. A news announcement that indicates a macroeconomic shift that was already anticipated by market participants, is hardly any news at all.

It therefore seems important to consider textual and numeric information jointly to capture contextual meaning correctly. A few of the above mentioned NLP methods allow for such joint, multimodal data processing. However, more work is still needed to better integrate natural language analyses into established economic and financial models. In general terms, if we have target variable $\boldsymbol{y}$, corpus $\boldsymbol{\mathcal{T}}$, and numeric data $\boldsymbol{X}$, we want to use a model that performs the parameter learning for both the function for the text feature representation $g()$ with parameters $\Omega_g$ and the overall supervised model fitting $f()$ with parameters $\Omega_f$ optimally with respect to a

specified prediction loss function between target $\boldsymbol{y}$ and prediction $\hat{\boldsymbol{y}}$ such that

$$\arg\min_{\boldsymbol{\Omega}} \text{loss}\left(\boldsymbol{y}, \hat{\boldsymbol{y}}\right) = \text{loss}\left(\boldsymbol{y}, f(\boldsymbol{X}, g(\boldsymbol{\mathcal{T}}|\boldsymbol{X}; \boldsymbol{\Omega}_g); \boldsymbol{\Omega}_f\right). \tag{2.38}$$

For instance, a relatively straightforward choice for those functions to capture both text and numeric data, can be a word count model that adds the numeric features to the text feature matrix that contains the term frequencies. Feature regularization can be applied to not penalize important numeric inputs if economic theory required their inclusion. Furthermore, in the area of topic modelling, several methods have been suggested to handle additional numeric meta-data to improve topic estimations themselves as well as associated downstream tasks such a regression or classification. Supervised topic models that can handle covariates are for example STM (Roberts et al., 2016) and DOLDA (Magnusson et al., 2020). Topics models in the spirit of STM incorporate document metadata, but in order to better predict the content of documents rather than to predict an outcome. SCHOLAR (Card et al., 2018) is a supervised topic model that generalises both sLDA (Blei and McAuliffe, 2008) as it allows for predicting labels, and SAGE (Eisenstein et al., 2011) which handles jointly modelling covariates via 'factorising' its topic-word distributions into deviations from the background log-frequency of words and deviations based on covariates. SCHOLAR is solved via neural variational inference (Kingma and Welling, 2014; Rezende et al., 2014). In Chapter 5, we introduce Bayesian Topic Regression (BTR) as well as rSCHOLAR (Ahrens et al., 2021) that we published at EMNLP 2021. rSCHOLAR is an extension of SCHOLAR with regression layer to allow for regression tasks. BTR is designed with applications for causal inference in mind. Over recent years, research on multimodal data fusion has picked up in popularity in the NLP community. Erickson et al. (2020) introduce an extensive auto-machine learning (AutoML) suite for multimodal learning, particularly for fusing text and numeric data. Baltrusaitis et al. (2019) provide an extensive survey on multimodal machine learning and data fusion. An area that seems crucial for advancements in economic and financial modelling – yet it hasn't been too deeply and systematically explored in this domain thus far. Future research into multimodal NLP modelling for economics and finance might yield promising advancements.

## 2.5 Challenges for NLP in Economics and Finance

Concluding this NLP background chapter, we point out the biggest complexities and challenges for NLP in economics and finance.

1. **Low frequency data problem**: particularly macroeconomics and marcofinance datasets often contain hundreds or a few thousands of data points instead of millions or billions as in classic NLP datasets.

2. **Long document problem**: Each data point is often associated with rather long text sections (thousands or ten-thousands of tokens) instead of shorter text sequences (token length: 100-500) found in many classic NLP datasets.

3. **Specific language and jargon**: financial and economic terminology is not necessarily guaranteed to be captured well by neural language models trained on Wikipedia and general book libraries.

4. **Multimodal data problem**: many text datasets in economics and finance come accompanied by potentially relevant numeric metadata. For instance, economic reports and financial statements are usually written in the context of current market conditions. Without the economic and financial context that is often encoded in numeric data, text passages can be ambiguous or misleading in their meaning.

5. **Numeracy**: Closely related to the multimodal data problem. The message or sentiment in economic and financial texts, oftentimes crucially depends on the numbers within them. A 1 percent increase in inflation is a very different story than a 100 percent increase in inflation.

None of these problems have been fully solved. Neither would it be realistic to assume that a single research contribution could solve them all. Progress has been made in many of those areas. For example, financial transformers such as FinBERT (Araci, 2019) and FLANG-BERT (Shah et al., 2022) capture financial domain jargon measurably better than non-financial transformer models (see Chapter 4). The importance of the multimodality aspect has been stressed by Das et al., yet more work in this direction is still needed. This list of challenges is meant to provide

an overview about where additional and concerted research work is particularly needed in the intersectional field of NLP for economic and financial modelling.

# 3 Background - NLP for Economics and Finance

This chapter provides the foundational methodological and literature background on monetary policy research with a particular focus on the analysis of monetary policy shocks, central bank communication, and the role of NLP methods for it.

## 3.1 NLP for Monetary Economics

Understanding the (causal) effect of monetary policy on economic and financial variables is one of the most fundamental empirical questions in monetary economics. An extensive branch of literature focuses on this question; Christiano et al. (1999) and Ramey (2016) provide surveys on the history and development of the research field as well as an extensive overview of the most prominent monetary policy shock identification methods. In this section, we first outline how a 'monetary policy shock' can be defined. We then point out the key identification challenges and the most commonly used methodological approaches to date. We conclude this section with the identification of research gaps in the literature and where this thesis aims to contribute.

### 3.1.1 What is a Monetary Policy Shock

The starting point of any attempt to understand the (causal) macroeconomic effects of monetary policy is to identify the unanticipated, exogenous element in monetary actions to avoid endogeneity concerns. In the monetary economics literature, the terms of monetary policy *shocks*, *innovations*, or *instruments* can have different meanings (Ramey, 2016). A *shock* might be considered as a primal, unforeseen disruption to the economy, which since unanticipated, affects the information set and expectation formation of market agents (Bernanke, 1986). However, modern day monetary policy usually follows clear and rather transparent rules, which make primal monetary policy *shocks* to the economic system less meaningful in the sense of economic

theory (Ramey, 2016). There are on the other hand unforeseen deviations from monetary policy makers' rules, which, if correctly identified, can be regarded as exogenous, unanticipated policy changes to the economic system. Depending on the identification methods, these are either defined in the realm of economic theory as *innovations* (i.e. unexplained variances when regressing monetary policy targets on their respective monetary policy rules) or as *instruments* (i.e. proxy measures to identify the exogenous, unanticipated element of a policy decision). What a *shock*, an*innovation*, and an *instrument* all have in common, is that they aim to capture the unexpected or unexplainable part in monetary policy decisions – loosely speaking, monetary policy *news* to the market. In this thesis, we also use the term *shock* when we more simply mean any form of monetary policy *news*, and we shall use these two terms interchangeably.

There are two overarching identification challenges that such shocks need to overcome (see also, Ramey, 2016):

1. **Exogeneity problem:** They need to be exogenous with respect to other relevant contemporary or lagged variables in the economic system. This ensures that we don't measure a reaction of the central bank to changes in the economy, but rather a reaction of the economy to a central bank policy change. This exogeneity requirement is often imposed in many economic models by what is often called *recursiveness* assumptions (Christiano et al., 1999).

2. **Foresight problem:** The policy change must not already be anticipated by market agents, as otherwise they would already have reacted (or priced in) the future change in policy into their reaction and expectation functions.

### 3.1.2 Overview of Identification Methods

Different classes of conceptual and methodological identification frameworks have been suggest in recent years and decades. We outline the most prominently used methods in modern monetary policy research.

#### 3.1.2.1 Vector Autoregressive Models

A commonly used shock identification framework in monetary economics are structural vector autoregressive (SVAR) models (Christiano et al., 1999; Ramey, 2016).

The aim is to model the dynamics of several key macroeconomic processes and their interaction with each other. As a stylized example (loosely based on Ramey, 2016), let there be a system

$$\Theta_{1,t} = \omega_{1,2}\Theta_{2,t} + \omega_{1,3}\Theta_{3,t} + \epsilon_{1,t} \tag{3.1}$$

$$\Theta_{2,t} = \omega_{2,1}\Theta_{1,t} + \omega_{2,3}\Theta_{3,t} + \epsilon_{2,t} \tag{3.2}$$

$$\Theta_{3,t} = \omega_{3,1}\Theta_{1,t} + \omega_{3,2}\Theta_{2,t} + \epsilon_{3,t} \tag{3.3}$$

where $\Theta_{1,t}$ includes all relevant real economic variables (and their lags) such as output, consumer prices index, unemployment and so on. $\Theta_{2,t}$ is the federal funds rate. $\Theta_{3,t}$ represents the monetary stock measure (e.g. M1, M2). The central bank's monetary policy rule would therefore be equation 3.2 in which the monetary policy shock would be represented by $\epsilon_{2,t}$ (Ramey, 2016).

For the identification of $\epsilon_{2,t}$, Christiano et al. (1999) then introduced what is now coined as *recursiveness* assumptions, which imposes a certain economic theory driven structure into the VAR model. Such recursiveness assumptions are that contemporaneous values of the macroeconomy $\Theta_{1,t}$ affect the monetary policy decision making process at time $t$, i.e. $\omega_{2,1} \neq 0$. However, monetary stock and reserve measures do not enter the monetary policy decision rule, hence $\omega_{2,3} = 0$. The final recursiveness assumption they make in order to be able to identify the effect of the monetary policy shock, is that none of the contemporaneous macroeconomic variables ($\Theta_{1,t}$) will react in period $t$ to the monetary policy decisions nor the monetary stock dynamics. That is, $\omega_{1,2} = \omega_{1,3} = 0$.

An extension to the above described SVAR approach are factor-augmented vector autoregressive (FAVAR) models (Bernanke et al., 2005). FAVAR models are similar in in their setup to the SVAR models, however, they can usually contain hundreds of economic and financial factors (Ramey, 2016) in order to control for endogenous macroeconomic and macrofinancial dynamics when trying to identify monetary policy shocks.

However, both the SVAR and the FAVAR model heavily rely on two strong assumptions. First, that contemporaneous economic variables do not respond the the monetary policy shock within the assessment period. This is a particularly strong assumption especially when the time periods $t$ pertain to monthly or even quarterly data. This assumption stands in strong contrast to established findings about responses and interactions among macroeconomic variables and

monetary policy, as brought forward for instance through dynamic stochastic general equilibrium (DSGE) models (Smets and Wouters, 2007), which are these days widely used by central banks to model the general macroeconomic market dynamics.

Second, VAR models hinge on the assumption that there is no foresight problem on the side of the the policy maker. That is, it is assumed that the central bank does not have superior, private information when it comes to its policy decision making process. This is certainly a strong assumption. Barth III and Ramey (2001) include Fed's Greenbook forecasts in their VAR equations. However, their findings were not able to reconcile the prize puzzle that such structural VAR models usually predict and which runs counter to economic theory. That is, such models predict an increase in inflation after unanticipated tightening of monetary policy.

Finally, such VAR models use by and large purely numeric data on monthly or quarterly frequency. Especially when it comes to modern day monetary policy steering, central bank communication is regarded as a key policy tool. Further work is needed to adjust such models to properly capture measures of central bank communication and to be able to make statements about communication effects.

#### 3.1.2.2 Narrative Models

A different yet prominent shock identification method is the *narrative approach*. It has been applied for the identification of various different policy shocks (Ramey, 2016). In the field of monetary policy, Romer and Romer (1989, 2004) contributed seminal papers. Despite its name, this approach is not necessarily focusing on linguistic pattern identification methods or on any approaches from NLP. Narrative methods focus on identifying information in historical documents that provides evidence for changes in, say, central bank internal monetary policy assessments and deliberations that reflects exogenous and unanticipated shifts in monetary policy decisions. Researchers usually manually screened policy documents to create such shock series. Ultimately, central bank internal information, such as Greenbook forecasts and the manually created shock series, is used to model the development of the target variable, which is usually the central bank's key interest rate (Romer and Romer, 2004). The series of regression residuals, that is, the unexplained variance in that model, is then labelled as the monetary policy shock. This shock series is then in turn inserted into a structural VAR model to assess

it's impact on macroeconomic variables. Similar constraints apply as in the above described case for the SVAR and FAVAR models. The setup aims to identify shocks that are purely exogenous and unanticipated by market participants. In particular, it aims at controlling for the private and potentially superior information a central bank possess when it enters its policy deliberation process. However, the Romer and Romer (2004) approach, still leads to a 'prize puzzle' where contractionary monetary policy has expansionary effects on macroeconomic variables when modern data series beyond the 2000s are considered (Ramey, 2016). Ahrens (2018) suggests that the the Romer and Romer shock series is likely not capturing the entire private information set of the Fed. By using topic modelling methods on additional briefing documents, the Beigebooks, endogenous central bank reactions to macroeconomic developments can be better controlled for and the prize effect can be remedied (Ahrens, 2018). Hansen and McMahon (2016) employ text analysis methods to measure and distinguish the information in FOMC statements pertaining to the state of the economy and the information concerning forward guidance. Using a FAVAR framework, they find that forward guidance shocks have a stronger effect on macroeconomic variables than shocks on the current and future economic outlook. Neither of the shocks are considered substantial. However, in the end such narrative methods ultimately rely on above described VAR analyses and therefore the same drawbacks apply. Particularly the reliance on monthly or quarterly data and the recursiveness assumption might be difficult to justify, as previously argued.

#### 3.1.2.3 High Frequency Identification Models

An alternative approach that, in theory, does not require such strong structural recursiveness assumptions, makes use of high frequency data to measure market reactions immediately before and after a central bank announcement has been made. Prominent papers include Kuttner (2001); Cochrane and Piazzesi (2002); Gürkaynak et al. (2005); Piazzesi and Swanson (2008); Barakchian and Crowe (2013); Gertler and Karadi (2015); Hattori et al. (2016); Nakamura and Steinsson (2018); Cieslak and Schrimpf (2019); Ehrmann and Talmi (2020); Leombroni et al. (2021); Gómez-Cram and Grotteria (2022). The general finding is that monetary policy shocks have relevant effects on a broad class of asset classes as well as on market expectations of the yield curve of the target interest rates.

This type of event study approach allows to relatively cleanly identify monetary policy shocks as it focuses on those short time intervals when monetary policy changes are made public to the market. Importantly here, the shocks are identified indirectly. A shock is considered to have happened, when markets have reacted 'sufficiently strongly' around an announcement date. However, most of the above mentioned studies ignore the actual textual content provided by the central bank, which does not allow more nuanced analyses of actual central bank communication effectiveness. A few exceptions exist. Nesbit (2020) proposes a word count based instrumental variable framework to identify monetary policy shocks in FOMC transcripts. Ehrmann and Talmi (2020) measure textual differences between central bank announcements and find that higher levels of textual similarity to the previous announcement statement are usually associated with lower market volatility after the announcement date. Hansen et al. (2019) analyse the Bank of England's Inflation Reports via topic modelling and find that communication plays an important role in shaping perceptions of uncertainty in long-run interest rates. They use a 1-day window in their event study analysis. Ellen et al. (2022) construct a monetary policy shock series based on the difference in narrative focus between central bank statements and news media coverage for the example of Norway. Following a high frequency identification approach, they confirm earlier findings in the literature that monetary policy shocks cause measurable macroeconomic responses. They furthermore highlight the pivotal role of news media as information intermediaries and catalysts in the process of forming market expectations. Gómez-Cram and Grotteria (2022) apply a video analysis on words mentioned during central bank press conference videos and align high-frequency financial data with the time of the mentioned words. They find that the most substantial asset price movements during press conference periods closely track time periods in which the central bank chair clarifies new or altered policy statements, or periods in which updates on forward guidance are addressed. Finally, Neuhierl and Weber (2019) assess the tone of US Fed chair and vice-chair speeches, again based on word count frequencies, and find that it can explain negative stock market price dynamics.

### 3.1.3 Further Research in NLP for Monetary Economics

The analysis of the actual textual content of central bank documents and speeches, and the usage of systematic NLP methods for identifying monetary policy shocks is still a relatively nascent development. At the same time, central bank communication has manifested itself as a key policy making tool to steer markets and their expectations and it is here to stay (Blinder, 2018; Draghi, 2017) – not just since the heightened focus on forward guidance following the global financial crisis where the zero lower bound on key interest rates constrained classical monetary policy steering channels. Haldane and McMahon (2018) outline the importance of central banks' roles in shaping public narrative on economic conditions and uncertainties.

However, essentially all above described approaches and analyses focus on changes in actual policy rates as part of the identification strategy. Empirically, though, there is very little surprise in monetary policy announcements (Gorodnichenko and Weber, 2016). The average change in the Fed Fund futures market around announcements since the 1990s is only 2.7 basis points. To the extent that central bank communication between meetings shapes expectations of subsequent policy decisions, it may contain the true surprises. What is more, a pure focus on central bank meetings only provides a rather infrequent updating cycle.

We identify a clear need for (i) the creation of empirical datasets that systematically capture the continuous flow of central bank communication, e.g. in terms of central bank speeches addressing the markets, (ii) the development and test of multimodal NLP methods that are suitable for a data environment of relatively few (hundreds to thousands) data points, however each of them containing rather long text corpora (several thousand words) as well as potentially relevant numeric metadata.

# 4 An NLP Benchmark for Economics and Finance

In this chapter, we introduce our work towards establishing an NLP benchmark foundation for economics and finance.

**Authors:** Maximillian Ahrens[1], Michael McMahon

**Publication venue**: Working Paper; presented at CEPR Webinar for Central Bank Communication[2]

## 4.1 Abstract

We introduce EcoFinBench, a natural language processing (NLP) benchmark suite for the domains of economics and finance. We comprehensively test a large array of NLP models across multiple domain specific datasets for sentence classification. Specifically, we evaluate dictionary models, word count models, topic models, and modern transformer models. Furthermore, we introduce two new datasets to the research community. The Bluebook dataset for text-only sentiment analysis in monetary policy, and the Greenbook dataset for multimodal sentiment analysis.

We focus on datasets that require the models to work with relatively few data points and long average text lengths - typical characteristics of datasets in the economic and financial domain. From our findings, we conclude that particularly in the multimodal domain, more research is needed. Furthermore, dictionary models – still widely used as a default text analysis tool in economics and finance – underperform substantially across all evaluated datasets. With our benchmark suite we aim to lay the foundation for a systematic assessment on the most

[1] main author
[2] https://www.youtube.com/watch?v=DiJQMzytKQY

commonly used NLP models in economics and finance. The current benchmarking task focuses on sentence classification of which sentiment analysis is likely the most widely applied use case in economics and finance. This is why we chose it as the starting point for our benchmarking framework. We aim to extend our benchmark to include additional tasks such as document similarity or named-entity-recognition (NER) in the future. To our knowledge, we are the first to provide such NLP benchmarking assessment for economics and finance.

## 4.2 Introduction

Text analysis has a long tradition in economics and finance. Tasks such as financial sentiment analysis, text classification, document similarity assessment, and information extraction from text documents play a pivotal role in these domains. Financial dictionary models, in particular Loughran and Mcdonald (2011) have for a long time been a workhorse model used by both practitioners and researchers in this domain, despite or perhaps because of its relatively simplistic conceptual framework.

As the field of natural language processing (NLP) progressed, new text analysis methods found increasing application in economics and finance. Topic models witnessed and still witness substantial popularity (Hansen and McMahon, 2016; Hansen et al., 2018; Larsen and Thorsrud, 2019; Ahrens and McMahon, 2021). With the introduction of transformers (Vaswani et al., 2017), deep neural network models for NLP found their way into economics and finance research. Transformers such as BERT (Devlin et al., 2018) became the foundational models for financial domain adaptations such as FinBERT (Araci, 2019) and FLANG-BERT (Shah et al., 2022).

Advancements in NLP have benefited from concerted actions to establish common benchmark datasets, tasks, and evaluation metrics in the research community - prominent examples being the General Language Understanding Evaluation (GLUE) (Wang et al., 2019b) and SuperGLUE (Wang et al., 2019a). However, there is a lack of an established NLP benchmark for the domains of economics and finance, as pointed out by Ash et al. (2021). Shah et al. (2022) might be the first paper to address this benchmark vacuum, suggesting the Financial Language Understanding Evaluation (FLUE) benchmark tasks. But their work considers transformer architectures only. They compare BERT (Devlin et al., 2018) and ELECTRA (Clark et al., 2020) to the authors' domain adapted equivalents. What is needed is a holistic benchmark, comparing the classes of NLP models that have been introduced into and adopted by the economics and finance domain

over the past years. It is important to take stock, systematically evaluate, and produce visibility about which NLP models work best under which dataset characteristics in these fields.

As one specific case in point, in economics and finance, text data often comes accompanied with important numeric metadata. For instance, economic reports and financial statements are usually written in the context of current market conditions. Without the economic and financial context that is often encoded in numeric data, text passages can be ambiguous in their meaning. Take for example the statement of a central bank that forecasts the inflation rate to increase by 2 percentage points until the end of the year. Is this good, bad, or neutral news for the financial markets? It depends. It depends on the current economic conditions, such as the current inflation rate. If such statement was made in a deflationary economic environment, in which the consumer price index (CPI) is substantially below the central bank's target (of usually 2%), a signal to the markets that CPI is about to pick up could be good news. On the other hand, the same statement given in an economic environment of hyperinflation would be seen as bad market news. In short, we need multimodal benchmark evaluations in which we test models on how well they can make predictions that rely on the joint assessment of text and numeric data. In the future, we might want to consider adding further modalities such as audio, image, or video.

The general motivation for an NLP benchmarking framework in economics and finance goes of course beyond the above example. NLP methods have found their way into various research projects and experiments in these domains. We use such NLP methods to be able to analyse new datasets and to answer new research questions – or to answer existing research questions with new insights. For economics and finance researchers, NLP methods are usually one tool in the econometric methodology toolbox. As such, we ought to ensure that we use the best and most suitable analysis tool and, even more importantly, that our results, conclusions, and policy recommendations are based on the true underlying data generation processes and not on an artifact of a singular model configuration. In short, our results derived from text data analysis should be robust across NLP models. For that, we need to examine which classes of NLP models work best for which downstream tasks and for which dataset characteristics. But even then, we might only be able to derive guidelines of which NLP models classes tend to do well in which use cases. We believe it would be good practice for any NLP-centered work in the economics

and finance domain to consider an array of NLP models to add further robustness to research findings whilst reducing model bias in empirical findings. Beyond model bias, we also need to be wary of information leakage, which can be a substantial problem when using modern large language models. Oftentimes, it is not clear on what datasets LLMs have been trained. This might lead to the issue that a researcher's test set had been part of the LLM's training set. Or, a somewhat subtler version of this, that the LLM has not been directly trained on a researcher's test set, however, the LLM has been trained on data lying in the future of the researcher's test set. And hence, the model could have indirectly learned related world knowledge that would not have been accessible at the real-time evaluation of the test set. Particularly for time-series analyses in economics and finance, such 'foresight' data leakage can easily invalidate test results. It is overall paramount for responsible and credible NLP research in economics and finance that issues such as (but not limited to) potential model bias and information leakage are being carefully investigated and addressed. In our benchmark, we do point out data leakage risks when an LLM has already potentially been trained on some of our test sets. In such cases, we also report the test results from the original papers to avoid reporting 'test' results based on potential training data.

With this paper, we aim to provide a starting ground for building a systematic NLP benchmark evaluation in economics and finance. We establish an initial NLP model benchmark suite that includes dictionary models, word count models, topic models, and large language models (LLM). An overview on the model suite is given in Section 4.5. We actively welcome contributions and additions from the research community. We chose some of the most common NLP datasets used in economics and finance and added an additional multimodal NLP dataset to establish a dataset foundation to which the community is invited to contribute. The current benchmarking task is sentence classification of which sentiment analysis is likely the most widely applied use case in economics and finance. This is why we chose it as the starting point for our benchmarking framework. An overview on the dataset suite is provided in Section 4.4. We aim to extend our benchmark to include additional tasks such as document similarity or named-entity-recognition (NER) in the future.

Finally, we suggest that it might be time for the economics and finance community to move on from using hand-crafted dictionary models such as Loughran and Mcdonald (2011) (LMdict)

as their default methods. Our findings show that LMdict methods underperform substantially across all evaluated datasets.

### 4.3 Related Work

In their recent paper, Ash et al. (2021) point out the lack of established NLP benchmarks in economics and finance. Recently, several papers compared particular subsets of NLP models used in economics and finance on various downstream tasks. Das et al. (2022) compare the workhorse dictionary model from Loughran and Mcdonald (2011) against the authors' machine learning based financial dictionary models. Overall results don't show substantial performance increases by the new approaches. Boukes et al. (2020) and van Atteveldt et al. (2021) find that dictionary model performance is often close to chance and inter-dictionary consistency is low. Such findings suggest that it might be time to default to more performant NLP model alternatives. Our paper features a dedicated assessment of the performance of the workhorse dictionary models by Loughran and Mcdonald (2011) against data driven machine learning methods.

Leippold (2023) performs a benchmark evaluation of adversarial attacks on financial texts created with GPT-3 (Brown et al., 2020) and finds that standard dictionary models are substantially less robust to such attacks than context-aware transformers such a models from the BERT (Devlin et al., 2018) family. In contrast, we compare a large array of NLP models that includes dictionary models, transformer models, and various other NLP models popular in economics and finance. We furthermore assess model performances across various text-only and multimodal (text + numeric data) datasets. The importance of multimodal task benchmarks has been recently emphasized by Ash et al. (2021).

Shah et al. (2022) propose FLUE, a transformer-only benchmark for the financial domain. However, what is particularly needed in this domain is an evaluation that compares the entire spectrum of NLP models, from simple dictionaries and word count models to state-of-the-art (SOTA) LLMs. We are laying the foundation for such a benchmark suite.

## 4.4 Datasets and Evaluation

### 4.4.1 Datasets

Our current benchmark suite contains the following datasets for which Table 4.1 shows the descriptive statistics:

1. The Financial Phrase Bank (FPB) dataset[3] (Malo et al., 2014), contains sentences from financial news that are annotated as reflecting either positive, negative, or neutral sentiment. We use only sentences that have 100% annotator agreement.

2. The Twitter Financial News (TFN) dataset[4] contains annotated finance-related tweets classified as either bearish, bullish, or neural.

3. The FOMC Bluebook Alternatives (FBA) dataset[5] contains FOMC statement alternatives and their respective fed funds rate (FFR) decision, which is used as the label. The original labels are discrete. We categorise the label as negative if the FOMC decision leads to a decrease in the FFR, neutral if the FFR is unchanged, and positive if the FFR has been increased.

4. The FOMC Greenbook CPI forecasts (FGC) dataset[6] contains the Greenbook paragraphs about inflation (consumer price index) and the associated one-quarter-ahead Greenbook forecast that we use as a label. We define the forecast as either increasing, decreasing, or unchanged. Furthermore, the dataset contains numeric metadata which consists of the one-quarter-ahead forecasts on CPI, GDP growth, and unemployment of the previous FOMC meeting.

Table 4.1: Descriptive statistics of datasets in EcoFinBench

| dataset | type | total | train | val | test | neg | neut | pos | mean len | max len | min len |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Fin Phrase Bank | text | 2,264 | 60% | 15% | 25% | 13% | 61% | 25% | 122 | 315 | 9 |
| Fin. Twitter | text | 11,931 | 64% | 16% | 20% | 15% | 20% | 65% | 86 | 227 | 2 |
| Bluebooks | text | 418 | 64% | 16% | 20% | 6% | 75% | 17% | 2,716 | 5,934 | 666 |
| Greenbooks | text+tab | 144 | 64% | 16% | 20% | 48% | 6% | 47% | 3,940 | 13,063 | 292 |

[3] https://huggingface.co/datasets/financial_phrasebank
[4] https://huggingface.co/datasets/zeroshot/twitter-financial-news-sentiment
[5] https://github.com/mcmahonecon/FOMC-Alt-Statements
[6] Author Michael McMahon. Available upon request

The datasets we chose reflect what we think are some of the key characteristics of text sources in the domain of economics and finance:

i. Economics and finance text datasets can often be rather small, containing only hundreds or thousands of data points compared to millions or billions of data points in 'classic' NLP datasets.

ii. Economics and finance text datasets can often contain rather long text sequences, as reflected in the Bluebook and Greenbook dataset which have mean sequence lengths of around 3,000 and 4,000 words, and maximum sequence lengths of 6,000 to 13,000 words.

iii. Economics and finance text datasets can often contain potentially relevant numeric metadata that is vital to gauge the context in which a statement is to be interpreted, as reflected in the Greenbook dataset.

We aim to consistently expand the selection of economic and financial datasets and are actively welcoming contributions from the research community. Finally, a note on models in the benchmark suite that have used some of those datasets for training. FinBERT and FLANG-BERT have been fine-tuned on FPB. As we don't have the exact train-test split that was used by the authors, we report the test results on FPB from the original papers (where available). We cross-check those results with our own test results - being mindful that those results might have been subject to training data leakage. Our test results for FinBERT and FLANG-BERT, however, show very comparable results to the figures reported in the respective original papers.

### 4.4.2 Evaluation Metrics

We use the F1 score as the overall performance metric for all classification tasks. For classification tasks with more than 2 classes, we report the macro F1 score. The F1 score is widely used as an evaluation metric for machine learning and NLP classification tasks. Particularly in unbalanced datasets it provides a more meaningful assessment of the model performance than, say, simple accuracy. The F1 score is the harmonised mean over precision and recall. Precision also known as positive predictive value is defined as

$$\text{precision} = \frac{\text{tp}}{\text{tp} + \text{fp}}. \tag{4.1}$$

Recall is also known as sensitivity and is defined as

$$\text{recall} = \frac{\text{tp}}{\text{tp} + \text{fn}}. \tag{4.2}$$

Here, tp represents the true positives, fp the false positives, and fn the false negatives. The F1 score is then

$$\text{F1} = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{\text{tp}}{\text{tp} + \frac{1}{2}(\text{fp} + \text{fn})}. \tag{4.3}$$

For multilabel classification ($> 2$ classes), the macro F1 score calculates the F1 metric for each class label (one-vs-rest) and then takes the arithmetic (unweighted) mean over the per-class F1 scores. The macro F1 score therefore treats all classes equally independent of their support.

## 4.5 Models

In this section, we briefly outline the NLP models in our benchmark evaluation.

### 4.5.1 Dictionary Models

We use the Loughran and Mcdonald (2011) financial dictionary (LMdict) as the representative dictionary model for our benchmark. It is a dictionary models widely used by both researchers and practitioners in the fields of economics and finance. LMdict provides hand-crafted word lists along seven categories: negative, positive, uncertain, litigious, strong modal, weak modal, and constraining. Typically, these seven LMdict features represent the percentage share of words in the input sequence found in the respective word list categories. In practice, researchers often focus only on the first four of these categories (Das et al., 2022). We deploy the software implementation from the Software Repository for Accounting and Finance[7] by the University of Notre Dame. We define three different LMdict approaches.

**LMdict-naive:** We label the most simplistic implementation LMdict-naive. This approach only uses the positive (*%positive*) and negative (*%negative*) feature category. Moreover, LMdict-naive does not learn any parameter weights in a data-driven way. It classifies a sequences as

[7] sraf.nd.edu/loughranmcdonald-master-dictionary/

negative, neutral, or positive, according to the following naive thresholds:

$$y = \begin{cases} 0, & \eta \geq 0.75 \\ 1, & 0.75 < \eta > 0.25 \\ 2, & \eta \leq 0.25 \end{cases} \tag{4.4}$$

where $\eta$ is the *negative-positive ratio* defined as

$$\eta = \frac{\%negative}{\%negative + \%positive}. \tag{4.5}$$

**LMdict-lin:** We use the seven word list features from the LMdict software implementation: *% negative*, *% positive*, *% uncertainty*, *% litigious*, *% strong modal*, *% weak modal*, *% constraining*. We add the additional features *# of words*, *# of digits*, *# of numbers*, which are also provided by the LMdict software implementation. We then feed these features into a logistic regression to learn the parameter weights for the features given the training dataset.

**LMdict-nonlin**: We use the same feature set as in LMdict-lin, but instead of a logistic regression, we fit a model zoo of non-linear machine learning methods on the respective training dataset. We use the AutoGluon (Erickson et al., 2020) machine learning model zoo (see Appendix A.1).

### 4.5.2 Word Count Models

We extract the word counts from the input text sequences and represent them in a document-term-matrix (DTM). The word counts are subsequently weighted by their tf-idf score. The tf-idf scaled word counts then serve as features in a linear and a non-linear classification model.

**WordCount-lin:** We use the tf-idf scaled word counts as features in a logistic regression model that uses elastic net regularization. The ratio between lasso and ridge regularization is a hyperparameter that is being tuned via cross-validation in the model training phase.

**WordCount-nonlin:** We use the same feature set as in WordCount-lin, but instead of a logistic regression, we fit a model zoo of non-linear machine learning methods on the respective training dataset. We use the AutoGluon machine learning model zoo (see Appendix A.1).

### 4.5.3 Topic Models

We fit unsupervised latent Dirichlet allocation (LDA) models (Blei et al., 2003) on the respective training datasets. We estimate LDA models with $K = [5, 10, 50, 100, 250, 500, 1000]$ topics. We then use the estimated topic parameters as features in a logistic regression.

### 4.5.4 Transformer Models

We use BERT (Devlin et al., 2018) as our reference architecture for transformer models. We use the following pre-trained Hugging Face[8] implementations and then fine-tune them on our respective training datasets:

**BERT-base:** The standard BERT-base model (Devlin et al., 2018) which has not been further pretrained for financial domain applications.

**FinBERT:** FinBERT (Araci, 2019) is a BERT model that has been further pre-trained on a financial corpus containing a subset of the Thomson Reuters Text Research Collection (TRC2), and then been fine-tuned on a subset of the Financial Phrase Bank dataset (Malo et al., 2014).

**FLANG-BERT:** FLANG-BERT Shah et al. (2022) is also based on a BERT model. It has been pre-trained on general English corpora from Wikipedia and BookCorpus as well as financial corpora from SEC EDGAR, Reuters, Bloomberg, Seeking Alpha, and Investopedia. Furthermore, the model makes use of preferential token masking for financial terms.

[8] https://huggingface.co/

### 4.5.5 Multimodal Model Extensions

We add model extensions to deal with multimodal tasks.

**Tab+:** We use the TabularPredictor model architecture from AutoGluon to fuse tabular numeric data with text data. This yields the same models as described above, but with a multimodal fusion capability. Such models are labelled with a "Tab+" prefix.

**Multimodal Transformer**: We add one additional model, which we name Multimodal Transformer. This model uses the *MultimodalPredictor* model architecture instead of *TabularPredictor* (Erickson et al., 2020). The difference is that this approach directly fuses multiple neural network models for the different modalities. For more details, see Erickson et al. (2020). For the text modality, the model uses a transformer architecture - we use BERT-base.

## 4.6 Results

Table 4.2 shows the results for the text-only datasets. Table 4.3 summarises the results for the multimodal Greenbook dataset. In these tables, we report the best and worst performing models as well as the ensembled model of the AutoGluon model zoo. A detailed report of all individual models can be found in Appendix A.2 and A.3. In Figure 4.1, we show the macro F1 score benchmark across all datasets and across the NLP model classes.

**Financial transformer models perform best on text-only datasets [Table 4.2]:** Overall, (financial) transformer models perform best on text-only datasets. The difference is by and large statistically significant apart from the FOMC Bluebook dataset. While a financial transformer achieved a higher average F1 scores than a non-financial transformer, the difference is not statistically significant at a 5% level, as both means lie within two standard deviations of one another. Interestingly, relatively simple WordCount-based models perform quite competitively - they perform as well as (financial) transformers on the FBA dataset and closely behind the transformers on the TFN and FPB dataset.

**Financial transformer models struggle on multimodal economics and finance datasets [Table 4.3]:** For the multimodal Greenbook CPI dataset, the transformer models are far from being among the top performing models. The best performing model is an AutoGluon ensembling model that uses the Laughran-McDonald text features as well as the numeric metadata. These differences are statistically significant. The Greenbook dataset contains very few observations ($< 200$). Furthermore, its respective text sequences are relatively long with an average length of $3,000$ words per document. Such data characteristics can be quite common for economic or financial text datasets and we observe that even finance-tuned large language models such as FinBERT and FLANG-BERT struggle with this dataset and yield no edge over non-financial transformers (BERT-base) on this dataset. The Multimodal Transformer performs in the same ballpark as the text-only transformers, unable to improve performance by leveraging the information in the numeric metadata. A possible reason for the subpar performance of transformer models on the might be the long sequence length of the dataset. For example, any model based on the BERT-architecture will have to truncate input sequences at 512 tokens. Since Greenbook sequences are on average even longer than Bluebook sequences, this means that even more information is lost. A maximum sequence length of 512 tokens is quite a strong limitation especially for a financial transformer, given that many real world datasets in the economic and financial domain contain substantially longer sequences (e.g. SEC filings and central bank documents). We aim to add transformer models that can handle longer text sequences, such as Longformer (Beltagy et al., 2020) Bigbird (Zaheer et al., 2020), which can handle up to 4096 tokens per input sequence. This should yield us additional insights whether sequencing length limitations might be the driving force behind the relatively weak performances of transformers on the multimodal dataset. However, the Bluebook dataset's average sequence length is with 2,700 words also much beyond the BERT cut-off. On that dataset, BERT models still rank among the best overall models. Sequence length limitations might therefore not be the only problem transformers are facing on the multimodal Greenbook dataset.

**Relatively simple word count models do very well across all datasets [Table 4.2]:** An interesting observation is that a relatively simple word count model performs quite well.

The linear word count model (WordCount-lin) performs on par with BERT-base on the FBA dataset. It also outperforms all alternative models on the TFN and FPB dataset apart from its non-linear word count counterparts and the considerably larger and more complex transformer models. On the multimodal dataset, models using fusing numeric data with word count features are also performing competitively, 8% (or 4 F1 score units) below the best model.

**The Loughran-McDonald dictionary model underperforms substantially across all datasets [Figure 4.2]:** Figure 4.2 compares the performance of approaches using LMdict features versus alternative NLP models. In particular, we compare LMdict-naive, LMdict-lin, and LMdict-nonlin against both FinBERT, WordCount-lin, and WordCount-nonlin.

In the text-only datasets, the LMdict-naive underperforms FinBERT between 34-76% (F1 score is 33-72 units lower), see Figure 4.2. LMdict-lin and LMdict-nonlin underperform FinBERT by 33-48% and 22-39%, respectively. And even in comparison to a linear word count feature model (WordCount-lin), all LMdict methods substantially underperform, even LMdict-nonlin. LMdict-naive underperforms WordCount-lin by 32-72%. LMdict-lin underperforms WordCount-lin by 30-40%. And LMdict-nonlin, the best non-linear machine learning model using LMdict features, underperforms WordCount-lin by 18-30%. All LMdict methods markedly underperform the non-linear word count model (WordCount-nonlin).

Similarly, in the multimodal dataset the LMdict methods lack behind in performance. LMdict-naive, LMdict-lin, and LMdict-nonlin underperform FinBERT by 47%, 20%, and 6% respectively. LMdict-naive and LMdict-lin also underperform WordCount-lin. Only LMdict-nonlin does marginally better than WordCount-lin. However, compared to the non-linear word count model (WordCount-nonlin), even the best performing LMdict model, LMdict-nonlin, underperforms by 26% (or 9.5 F1 score units).

**Unsupervised topic models show lack in performance across all datasets [Figure 4.1]:** Models using topic features from an unsupervised topic model (LDA) do not show competitive performance on sentiment analysis and sequence classification tasks. Current findings suggest that topic modelling strength relative to other models seems to correlate with sequence lengths in the respective dataset. For example, the topic modelling approaches do

not outperform the majority class baseline on the Twitter dataset, which also has the shortest average sequence lengths with 86 words per document. On the Bluebook dataset that has considerably longer average sequence lengths (2,700 words), the topic modelling approaches are performant, yet over 16 F1 score units (28%) behind the best models. On the Greenbook dataset, the performance distance to the best models is 16 F1 score units (45 %), yet they are on par with BERT or FinBERT. Unsupervised topic models have been quite frequently deployed for similar empirical research settings in economics and finance. The current results suggest that alternative models might yield better performances. We are currently working on adding non-linear machine learning models using topic features from unsupervised topic models. Equally, we are in the process of adding supervised topic models to obtain more nuanced empirical results on the competitive performance of topic modelling approaches (for sentiment analysis and sequence classification) in economics and finance.

Table 4.2: Benchmarks: text-only datasets

(a) **FOMC Bluebook Alternatives**
#train: 327 — #test: 82

| model | macro F1 |
|---|---|
| WordCount-nonlin-best | 96.4 (2.8) |
| FinBERT | 96.0 (5.0) |
| WordCount-nonlin-ens | 95.7 (3.7) |
| WordCount-lin | 92.4 (13.8) |
| FLANG-BERT | 91.0 (9.2) |
| BERT-base | 86.6 (12.6) |
| WordCount-nonlin-worst | 79.7 (5.4) |
| LDA-K1000 | 77.7 (8.2) |
| LDA-K500 | 76.8 (7.8) |
| LMdict-nonlin-best | 77.1 (7.7) |
| LDA-K250 | 75.9 (7.4) |
| LMdict-nonlin-ens | 75.4 (6.3) |
| LDA-K100 | 70.5 (8.1) |
| LDA-K50 | 67.8 (8.9) |
| LMdict-lin | 64.6 (6.6) |
| LMdict-naive | 63.3 (6.5) |
| LMdict-nonlin-worst | 54.4 (5.8) |
| LDA-K10 | 50.2 (10.6) |
| LDA-K5 | 40.1 (7.8) |
| Majority class | 28.8 (0.8) |

(b) **Twitter Financial News**
#train: 9,545, #test: 2,386

| model | macro F1 |
|---|---|
| FLANG-BERT | 82.2 (2.9) |
| FinBERT | 81.5 (1.3) |
| BERT-base | 80.8 (1.7) |
| WordCount-nonlin-best | 77.7 (0.0) |
| WordCount-nonlin-ens | 76.6 (0.0) |
| WordCount-lin | 67.8 (1.1) |
| LMdict-nonlin-best | 51.5 (0.0) |
| LMdict-nonlin-ens | 50.0 (0.0) |
| LMdict-nonlin-worst | 41.7 (0.0) |
| LMdict-lin | 37.3 (-) |
| LMdict-naive | 29.6 (-) |
| LDA-K100 | 26.4 (0.2) |
| LDA-K50 | 26.4 (0.0) |
| LDA-K10 | 26.4 (0.0) |
| LDA-K5 | 26.4 (0.0) |
| LDA-K1000 | 26.4 (0.0) |
| LDA-K500 | 26.4 (0.0) |
| Majority class | 26.4 (0.0) |
| WordCount-nonlin-worst | 26.4 (0.0) |
| LDA-K250 | 26.4 (0.0) |

(c) **Financial Phrase Bank benchmark**
#train: 1,698, #test: 566

| model | macro F1 |
|---|---|
| FinBERT* | 94.7 (1.2) |
| BERT-base | 94.4 (1.5) |
| FLANG-BERT* | 94.1 (1.5) |
| WordCount-nonlin-best | 87.1 (1.6) |
| WordCount-nonlin-ens | 87.0 (1.6) |
| WordCount-lin | 82.9 (1.8) |
| LDA-K1000 | 63.7 (4.0) |
| LMdict-nonlin-ens | 58.3 (2.2) |
| LMdict-nonlin-best | 58.3 (2.3) |
| LMdict-nonlin-worst | 52.9 (2.1) |
| WordCount-nonlin-worst | 51.8 (2.9) |
| LMdict-lin | 50.0 (-) |
| LDA-K500 | 47.6 (9.3) |
| LDA-K100 | 44.1 (4.0) |
| LDA-K250 | 44.0 (4.4) |
| LDA-K50 | 42.7 (3.2) |
| LDA-K10 | 29.6 (6.1) |
| LDA-K5 | 25.6 (2.2) |
| Majority class | 25.1 (-) |
| LMdict-naive | 23.0 (-) |

*The tables display average F1-score over 50 model runs with standard deviation in brackets, both in percent.*
**FinBERT and FLANG-BERT were fine-tuned*
*on the FPB dataset. The original F1 test score for FinBERT is 95%. The FLANG-BERT paper does not report F1 scores.*

Table 4.3: Benchmarks: multimodal dataset

FOMC Greenbook CPI Multimodal
#train: 112, #test: 28

| model | macro F1 |
|---|---|
| Tab+LMdict-nonlin-best | 46.7 (0.0) |
| Tab-best | 44.2 (0.0) |
| Tab+encod-best | 43.1 (0.0) |
| Tab+WordCount-nonlin-ens | 42.8 (0.0) |
| Tab+WordCount-nonlin-best | 42.7 (0.0) |
| WordCount-nonlin-best | 42.1 (0.0) |
| Tab+LMdict-nonlin-ens | 39.8 (0.0) |
| Tab-ens | 37.3 (0.0) |
| WordCount-nonlin-ens | 36.8 (0.0) |
| Tab+encod-ens | 34.8 (0.0) |
| BERT-base | 31.0 (7.1) |
| LDA-K100 | 30.6 (8.3) |
| LDA-K50 | 30.3 (8.0) |
| LDA-K10 | 29.9 (7.9) |
| LDA-K250 | 29.9 (7.6) |
| LDA-K1000 | 29.5 (5.6) |
| LDA-K500 | 29.5 (6.3) |
| FinBERT | 29.0 (7.2) |
| FLANG-BERT | 28.7 (6.6) |
| Multimodal Transformer | 28.1 (0.0) |
| LDA-K5 | 28.1 (7.6) |
| LMdict-nonlin-ens | 27.3 (0.0) |
| WordCount-lin | 25.6 (1.9) |
| LMdict-lin | 23.3 (0.0) |
| Tab+WordCount-nonlin-worst | 21.7 (0.0) |
| Tab+encod-worst | 21.7 (0.0) |
| Tab+LMdict-nonlin-worst | 21.7 (0.0) |
| Majority class | 21.7 (-) |
| Tab-worst | 21.7 (0.0) |
| Tab+LMdict-nonlin-worst | 21.5 (0.0) |
| Tab+WordCount-nonlin-worst | 21.1 (0.0) |
| LMdict-naive | 15.3 (0.0) |

*The table displays the average F1-score over 50 model runs with standard deviation in brackets, both in percent.*

Figure 4.1: F1 scores across all datasets, in percent

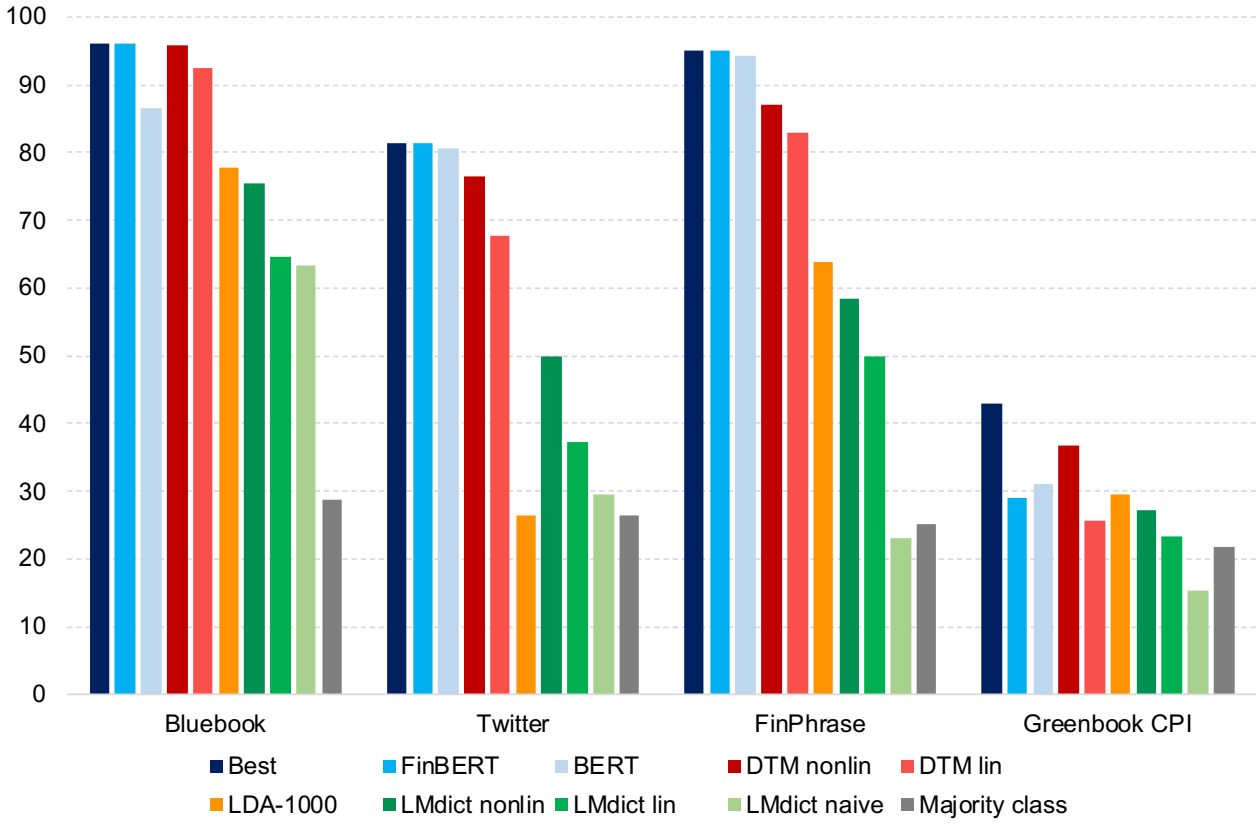
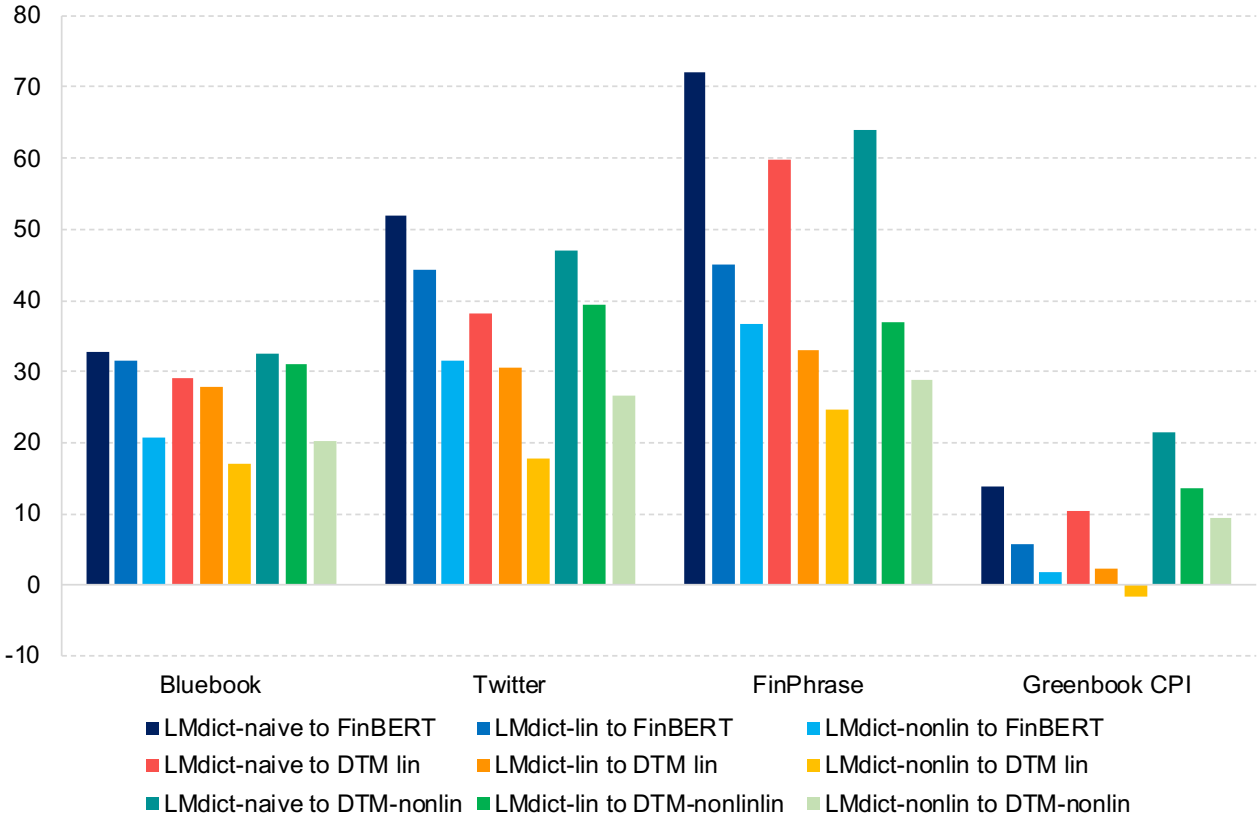



Figure 4.2: F1 score distances between dictionary model and other models, in percent



## 4.7 Discussion

In this paper, we introduce our work towards establishing a more comprehensive NLP benchmark evaluation for the domains of economics and finance. Over the past years, researchers introduced various NLP methods in these domains to be able to answer new research questions. Now might be a good time to take stock and start systematically evaluating which NLP methods work best for the most common domain specific tasks, given the most common domain specific dataset characteristics. In particular, we would like to underline the importance of evaluating multimodal (for now, text and numeric data) tasks, since many NLP datasets in economics and finance come accompanied by potentially relevant numeric metadata. The key findings of our initial benchmarking evaluation are:

1. Financial transformer models perform best on text-only classification datasets.

2. Financial transformer models struggle on multimodal economics and finance datasets, highlighting more need for concerted research efforts on fusing multimodal data and pre-training/fine-tuning models for tasks in these domains.

3. The Loughran and Mcdonald (2011) dictionary model - still widely used as a default text analysis tool in economics and finance - underperforms substantially across all evaluated datasets.

4. Relatively simple word count models do very well across all datasets.

Furthermore, we believe it would be good practice for any NLP-centered work in the economics and finance domain to consider an array of NLP models to add further robustness to research findings whilst reducing model bias in empirical findings. This benchmark showed that oftentimes relatively small and specialised models can still achieve competitive results compared to large language models. Especially when dealing with dataset characteristics that are common in macroeconomics and finance such as (1) relatively few data points (thousands vs millions), (2) relatively long text lengths (thousands of tokens per document), and (3) multimodality (text and numeric data), large language models do not always necessarily have the upper hand – at least not yet. However, large language models are becoming increasingly more powerful – for example as they can handle multimodality increasingly better and as their context window size

constantly increases. Simultaneously, the computational inference costs of such large language models keeps on decreasing as we make further research progress. It will likely be a wise research strategy to run an experiment with smaller and simpler NLP models initially in order to obtain a first approximation and baseline of the dynamics and complexities entailed in the specific data environment of interest. Such first insights can often guide the researcher to choose additional NLP model classes that seem promising for the respective task. We then recommend following an NLP model agnostic research approach, in which the researcher ultimately evaluates a zoo of models on a dedicated validation set of their data. Unless a researcher has strong arguments to do otherwise, we suggest to let the data speak which model seems to be best suited for a research question at hand. Furthermore, evaluating over a model zoo allows the researcher to run model robustness tests and the researcher can also opt to average (also known as *ensembling*) over a set models to avoid results being overly sensitive to individual model choices.

Our evaluation suite is by no means making a claim on being exhaustive. Our current benchmarking task is sentence classification of which sentiment analysis is likely the most widely applied use case in economics and finance. We plan to extend our benchmark to include additional tasks such as document similarity or named-entity-recognition (NER) in the future. With this paper, we aim to lay the foundational upon which the research community can easily build and extend. We highly encourage suggestions for additions of new models, datasets, tasks, and evaluation metrics. Going forward, we also aim to make our benchmark suite more interactive, providing a online interface that allows the research community to easily evaluate new datasets and models. We also plan to extend our benchmark suite beyond English, establishing a more comprehensive language coverage of both high and low resource languages.

# 5

# Bayesian Topic Regression For Multimodal Modelling and Causal Inference

In this chapter, we introduce our multimodal NLP algorithm, which is based on a supervised topic model.

**Authors:** Maximillian Ahrens[1], Julian Ashwin[2], Jan-Peter Calliess, Vu Nguyen

**Publication venue:** EMNLP 2021 [Ahrens et al. (2021)]

## 5.1 Abstract

Causal inference using observational text data is becoming increasingly popular in many research areas. This paper presents the Bayesian Topic Regression (BTR) model that uses both text and numeric information to model an outcome variable. It allows estimation of both discrete and continuous treatment effects. Furthermore, it allows for the inclusion of additional numeric confounding factors next to text data. To this end, we combine a supervised Bayesian topic model with a Bayesian regression framework and perform supervised representation learning for the text features jointly with the regression parameter training, respecting the Frisch-Waugh-Lovell theorem. Our paper makes two main contributions. First, we provide a regression framework that allows causal inference in settings when both text and numeric confounders are of relevance. We show with synthetic and semi-synthetic datasets that our joint approach recovers ground truth with lower bias than any benchmark model, when text and numeric features are correlated. Second, experiments on two real-world datasets demonstrate that a joint and supervised learning

[1]joined main author
[2]joined main author

strategy also yields superior prediction results compared to strategies that estimate regression weights for text and non-text features separately, being even competitive with more complex deep neural networks.

## 5.2 Introduction

Causal inference using observational text data is increasingly popular across many research areas (Keith et al., 2020). It expands the range of research questions that can be explored when using text data across various fields, such as in the social and data sciences; adding to an extensive literature of text analysis methods and applications (Grimmer and Stewart, 2013; Gentzkow et al., 2019). Where randomized controlled trials are not possible, observational data might often be the only source of information and statistical methods need to be deployed to adjust for confounding biases. Text data can either serve as a proxy for otherwise unobserved confounding variables, be a confounding factor in itself, or even represent the treatment or outcome variable of interest.

**The framework:** We consider the causal inference settings where we allow for the treatment variable to be binary, categorical or continuous. In our setting, text might be either a confounding factor or a proxy for a latent confounding variable. We also allow for additional non-text confounders (covariates). To the best of our knowledge, we are the first to provide such statistical inference framework.

Considering both text and numeric data jointly can not only improve prediction performance, but can be crucial for conducting unbiased statistical inference. When treatment and confounders are correlated with each other and with the outcome, the Frisch-Waugh-Lovell theorem (Frisch and Waugh, 1933; Lovell, 1963), described in Section 5.3.2, implies that all regression weights must be estimated jointly, otherwise estimates will be biased. Text features themselves are 'estimated data'. If they stem from supervised learning, which estimated the text features with respect to the outcome variable separately from the numeric features, then the resulting estimated (causal) effects will be biased.

**Our contributions:** With this paper, we introduce a Bayesian Topic Regression (BTR) framework that combines a Bayesian topic model with a Bayesian regression approach. This allows us to perform supervised representation learning for text features jointly with the estimation of regression parameters that include both treatment and additional numeric covariates.

In particular, information about dependencies between outcome, treatment and controls does not only inform the regression part, but directly feeds into the topic modelling process. Our approach aims towards estimating 'causally sufficient' text representations in the spirit of Veitch et al. (2020). We show on both synthetic and semi-synthetic datasets that our BTR model recovers the ground truth more accurately than a wide range of benchmark models. Finally, we demonstrate on two real-world customer review datasets - *Yelp* and *Booking.com* - that a joint supervised learning strategy, using both text and non-text features, also improves prediction accuracy of the target variable compared to a 'two-step' estimation approach with the same models. This does not come at a cost of higher perplexity scores on the document modelling task. We also show that relatively simple supervised topic models with a linear regression layer that follow such joint approach can even compete with much more complex, non-linear deep neural networks that do not follow the joint estimation approach.

## 5.3 Background and Related Work

### 5.3.1 Causal Inference with Text

Egami et al. (2018) and Wood-Doughty et al. (2018) provide a comprehensive conceptional framework for inference with text and outline the challenges, focusing on text as treatment and outcome. In a similar vein, Tan et al. (2014); Fong and Grimmer (2016) focus on text as treatment. Roberts et al. (2020); Mozer et al. (2020) address adjustment for text as a confounder via text matching considering both topic and word level features. Veitch et al. (2020) introduce a framework to estimate causally sufficient text representations via topic and general language models. Like us, they consider text as a confounder. Their framework exclusively focuses on binary treatment effects and does not allow for additional numeric confounders. We extend this framework.

**Causal inference framework with text:** This general framework hinges on the assumption that through supervised dimensionality reduction of the text, we can identify text representations that capture the correlations with the outcome, the treatment, and other control variables. Assume we observe iid data tuples $D_i = (y_i, t_i, \boldsymbol{W_i}, \boldsymbol{C_i})$, where for observation $i$, $y_i$ is the outcome, $t_i$ is the treatment, $\boldsymbol{W_i}$ is the associated text, and $\boldsymbol{C_i}$ are other confounding effects for which we have numeric measurements. Following the notational conventions set out in Pearl

(2009), define the average treatment effect of the treated (ATT[3]) as:

$$\delta = \mathbb{E}[y|\text{do}(t=1), t=1] - \mathbb{E}[y|\text{do}(t=0), t=1].$$

In the spirit of Veitch et al. (2020), we assume that our models can learn a supervised text representation $\boldsymbol{Z_i} = g(\boldsymbol{W_i}, y_i, t_i, \boldsymbol{C_i})$, which in our case, together with $\boldsymbol{C_i}$ blocks all 'backdoor path' between $y_i$ and $t_i$, so that we can measure the causal effect

$$\delta = \mathbb{E}[\mathbb{E}[y|\boldsymbol{Z}, \boldsymbol{C}, t=1] - \mathbb{E}[y|\boldsymbol{Z}, \boldsymbol{C}, t=0]|t=1].$$

Intuitively, to obtain such $\boldsymbol{Z_i}$ and consequently an unbiased treatment effect, one should estimate the text features in a supervised fashion taking into account dependencies between $\boldsymbol{W_i}$, $y_i$, $t_i$, and $\boldsymbol{C_i}$.

### 5.3.2 Estimating Conditional Expectations

To estimate the ATT, we need to compute the conditional expectation function (CEF): $\mathbb{E}[\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{Z}, \boldsymbol{C}]$. Using regression to estimate our conditional expectation function, we can write

$$\mathbb{E}[\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{Z}, \boldsymbol{C}] = f(\boldsymbol{t}, g(\boldsymbol{W}, \boldsymbol{y}, \boldsymbol{t}, \boldsymbol{C}; \Theta), \boldsymbol{C}; \boldsymbol{\Omega}). \tag{5.1}$$

Let $f()$ be the function of our regression equation that we need to define, and $\boldsymbol{\Omega}$ be the parameters of it. Section 5.3.4 covers text representation function $g()$. For now, let us simply assume that we obtain $\boldsymbol{Z}$ in a joint supervised estimation with $f()$. The predominant assumption in causal inference settings in many disciplines is a linear causal effect assumption. We also follow this approach, for the sake of simplicity. However, the requirement for joint supervised estimation of text representations $\boldsymbol{Z}$ to be able to predict $\boldsymbol{y}$, $\boldsymbol{t}$ (and if relevant $\boldsymbol{C}$) to be considered 'causally sufficient' is not constrained to the linear case (Veitch et al., 2020). Under the linearity assumption, the CEF of our regression can take the form

$$\boldsymbol{y} = \mathbb{E}[\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{Z}, \boldsymbol{C}] + \epsilon = \boldsymbol{t}\omega_t + \boldsymbol{Z}\boldsymbol{\omega_Z} + \boldsymbol{C}\boldsymbol{\omega_C} + \epsilon, \tag{5.2}$$

where $\epsilon \sim N(0, \sigma_\epsilon^2)$ is additive i.i.d. Gaussian noise, ie. $\mathbb{E}[\epsilon|\boldsymbol{t}, \boldsymbol{Z}, \boldsymbol{C}] = 0$ (see for example Angrist and Pischke (2008), chapter 3). Thus, $\sigma_\epsilon^2$ represents the conditional variance $Var(\boldsymbol{y}|\boldsymbol{t}, \boldsymbol{Z}, \boldsymbol{C})$.

[3] depicted is the ATT of a binary treatment. The same logic applies for categorical or continuous treatments.

The regression approximates the CEF. Hence, when the CEF is causal, the regression estimates are causal (Angrist and Pischke, 2008). In such a case, $\omega_t$ measures the treatment effect. Assuming that $\boldsymbol{Z}$ and $\boldsymbol{C}$ block all 'backdoor' paths, the CEF would allow us to conduct causal inference of $\boldsymbol{t}$ on $\boldsymbol{y}$ (Pearl, 2009).

We now shall revisit under which conditions, a decomposition of equation (5.2) into several separate estimation steps is permitted as described in the Frisch-Waugh-Lovell (or regression decomposition) theorem (Lovell, 2008), so that the regression estimates for $\omega_t$ remain unchanged and hence can still be considered as causal.

### 5.3.3 Regression Decomposition Theorem

The Frisch-Waugh-Lovell (FWL) theorem (Frisch and Waugh, 1933; Lovell, 1963), implies that the supervised learning of text representations $\boldsymbol{Z}$ and regression coefficients $\boldsymbol{\omega}$ cannot be conducted in separate stages, but instead must be learned jointly. The FWL theorem states that a regression such as in (5.2) can only be decomposed into separate stages, and still obtain mathematically unaltered coefficient estimates, if for each partial regression, we were able to residualize both outcome and regressors with respect to all other regressors that have been left out. In general, for a regression such as $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\omega} + \boldsymbol{\epsilon}$, we have a projection matrix $\boldsymbol{P} = \boldsymbol{X}(\boldsymbol{X}^\intercal\boldsymbol{X})^{-1}\boldsymbol{X}^\intercal$ that produces projections $\widehat{\boldsymbol{y}}$ when applied to $\boldsymbol{y}$. Likewise, we have a 'residual maker' matrix $\boldsymbol{M}$ which is $\boldsymbol{P}$'s complement $\boldsymbol{M} = \boldsymbol{I} - \boldsymbol{P}$. FWL says that if we could estimate

$$\boldsymbol{M}_{c,z}\boldsymbol{y} = \boldsymbol{M}_{c,z}\boldsymbol{t}\hat{\omega}_t + \hat{\epsilon}, \tag{5.3}$$

the estimates $\hat{\omega}_t$ of treatment effect $\omega_t$ in equations (5.2) and (5.3) would be mathematically identical (full theorem and proof in Section 5.3.3.1). Here, $\boldsymbol{M}_{c,z}$ residualizes $\boldsymbol{t}$ from confounders $\boldsymbol{C}$ and $\boldsymbol{Z}$. This is however infeasible, since $\boldsymbol{Z}$ itself must be estimated in a supervised fashion, learning the dependencies towards $\boldsymbol{y}$, $\boldsymbol{t}$ and $\boldsymbol{C}$. Equation (5.2) must therefore be learned jointly, to infer $\boldsymbol{Z}$ and the CEF in turn. An approach in several stages in such a setup cannot fully residualize $\boldsymbol{t}$ from all confounders and estimation results would therefore be biased. What is more, if incorrect parameters are learned, out of sample prediction might also be worse. We demonstrate this both on synthetic and semi-synthetic datasets (Section 5.6 and 5.7).

#### 5.3.3.1 FWL Theorem and Proof

The regression decomposition theorem or Frisch-Waugh-Lovell (FWL) theorem states that the coefficients of a linear regression as stated in equation (5.2) are equivalent to the coefficients of partial regressions in which the residualized outcome is regressed on the residualized regressors - this residualization is in terms of all regressors that are not part of this partial regression.

For a moment, let us assume there are no confounding latent (that is to be estimated) text features $\boldsymbol{Z}$. Our observational data only consist of outcome $\boldsymbol{y}$, our treatment variable $\boldsymbol{t}$ and other observed confounding variables $\boldsymbol{C}$,

$$\boldsymbol{y} = \boldsymbol{t}\omega_t + \boldsymbol{C}\boldsymbol{\omega_C} + \epsilon. \tag{5.4}$$

The FWL theorem states that we would obtain mathematically identical regression coefficients $\omega_t$ and $\boldsymbol{\omega}_C$ is we decomposed this regression and estimated each part separately, each time residualizing (ie. orthogonalizing) outcomes and regressors on all other regressors.

More generally, for a linear regression define

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

with $\boldsymbol{y} \in \mathbb{R}^{D\times 1}$, $\boldsymbol{\beta} \in \mathbb{R}^{K\times 1}$, $\boldsymbol{X} \in \mathbb{R}^{D\times M}$, which we could arbitrarily partition into $\boldsymbol{X}_1 \in \mathbb{R}^{D\times K}$ and $\boldsymbol{X}_2 \in \mathbb{R}^{D\times J}$ so we could also write

$$\boldsymbol{y} = \boldsymbol{X}_1\boldsymbol{\beta}_1 + \boldsymbol{X}_2\boldsymbol{\beta}_2 + \boldsymbol{\epsilon},$$

define projection (or prediction) matrix $\boldsymbol{P}$ such that

$$\boldsymbol{P} = \boldsymbol{X}(\boldsymbol{X}^\intercal\boldsymbol{X})^{-1}\boldsymbol{X}^\intercal. \tag{5.5}$$

$\boldsymbol{P}$ produces predictions $\widehat{\boldsymbol{y}}$ when applied to outcome vector $\boldsymbol{y}$,

$$\widehat{\boldsymbol{y}} = \boldsymbol{X}\widehat{\boldsymbol{\beta}} = \boldsymbol{X}(\boldsymbol{X}^\intercal\boldsymbol{X})^{-1}\boldsymbol{X}^\intercal\boldsymbol{y} = \boldsymbol{P}\boldsymbol{y}. \tag{5.6}$$

Also define the complement of $\boldsymbol{P}$, the residual maker matrix $\boldsymbol{M}$

$$\boldsymbol{M} = \boldsymbol{I} - \boldsymbol{P} = \boldsymbol{I} - \boldsymbol{X}(\boldsymbol{X}^\intercal \boldsymbol{X})^{-1}\boldsymbol{X}^\intercal \tag{5.7}$$

such that $\boldsymbol{M}$ applied to an outcome vector $\boldsymbol{y}$ yields

$$\boldsymbol{M}\boldsymbol{y} = \boldsymbol{y} - \boldsymbol{X}(\boldsymbol{X}^\intercal \boldsymbol{X})^{-1}\boldsymbol{X}^\intercal \boldsymbol{y} = \boldsymbol{y} - \boldsymbol{P}\boldsymbol{y} = \boldsymbol{y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}} = \hat{\boldsymbol{\epsilon}}. \tag{5.8}$$

**Theorem:**

The FWL theorem states that equivalent to estimating

$$\boldsymbol{y} = \boldsymbol{X}_1\widehat{\boldsymbol{\beta}}_{\mathbf{1}} + \boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\mathbf{2}} + \widehat{\boldsymbol{\epsilon}} \tag{5.9}$$

we would obtain mathematically identical regression coefficients $\widehat{\boldsymbol{\beta}}_{\mathbf{1}}$ and $\widehat{\boldsymbol{\beta}}_{\mathbf{2}}$ if we separately estimated

$$\boldsymbol{M}_2\boldsymbol{y} = \boldsymbol{M}_2\boldsymbol{X}_1\widehat{\boldsymbol{\beta}}_{\mathbf{1}} + \hat{\boldsymbol{\epsilon}} \tag{5.10}$$

and

$$\boldsymbol{M}_1\boldsymbol{y} = \boldsymbol{M}_1\boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\mathbf{2}} + \hat{\boldsymbol{\epsilon}} \tag{5.11}$$

where $\boldsymbol{M}_1$ and $\boldsymbol{M}_2$ correspond to the data partitions $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$.

**Proof of Theorem:**

This proof is based on the original papers (Frisch and Waugh, 1933; Lovell, 1963). Given

$$\boldsymbol{y} = \boldsymbol{X}_1\widehat{\boldsymbol{\beta}}_{\mathbf{1}} + \boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\mathbf{2}} + \widehat{\boldsymbol{\epsilon}} \tag{5.12}$$

left-multiply by $\boldsymbol{M}_2$, so we obtain

$$\boldsymbol{M}_2\boldsymbol{y} = \boldsymbol{M}_2\boldsymbol{X}_1\widehat{\boldsymbol{\beta}}_{\mathbf{1}} + \boldsymbol{M}_2\boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\mathbf{2}} + \boldsymbol{M}_2\widehat{\boldsymbol{\epsilon}}. \tag{5.13}$$

We obtain from equation (5.7) that

$$\boldsymbol{M}_2\boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\boldsymbol{2}} = (\boldsymbol{I} - \boldsymbol{X}_2(\boldsymbol{X}_2^\intercal\boldsymbol{X}_2)^{-1}\boldsymbol{X}_2^\intercal)\boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\boldsymbol{2}} = \boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\boldsymbol{2}} - \boldsymbol{X}_2\widehat{\boldsymbol{\beta}}_{\boldsymbol{2}} = \boldsymbol{0}. \tag{5.14}$$

Finally, $\boldsymbol{M}_2\hat{\boldsymbol{\epsilon}} = \hat{\boldsymbol{\epsilon}}$. $\boldsymbol{X}_2$ is orthogonal to $\boldsymbol{\epsilon}$ by construction of the OLS regression. Therefore, the residualized residuals are the residuals themselves. Which leaves us with

$$\boldsymbol{M}_2\boldsymbol{y} = \boldsymbol{M}_2\boldsymbol{X}_1\widehat{\boldsymbol{\beta}}_{\boldsymbol{2}} + \hat{\boldsymbol{\epsilon}} \quad \square. \tag{5.15}$$

The same goes through for $\boldsymbol{M}_1$ by analogy.

#### 5.3.3.2 $\mathbb{E}[\boldsymbol{y}|\boldsymbol{t},\boldsymbol{C}]$, where $\boldsymbol{t} \perp \boldsymbol{C}$, no $\boldsymbol{Z}$

In the simplest case assume there was no confounding text. Our observational data only consist of outcome $\boldsymbol{y}$, our treatment variable $\boldsymbol{t}$ and other potential confounding variables $\boldsymbol{C}$. The conditional expectation function is $\mathbb{E}[\boldsymbol{y}|\boldsymbol{t},\boldsymbol{C}]$. We can estimate it via one joint regression as

$$\boldsymbol{y} = \boldsymbol{t}\omega_t + \boldsymbol{C}\omega_C + \epsilon_0. \tag{5.16}$$

Now, assuming that the linearity assumption is correct, the fact that $\boldsymbol{t} \perp \boldsymbol{C}$ implies that $\boldsymbol{C}$ is not actually a confounder in this setup. We would obtain the exact same regression coefficient estimates for $\omega_t$ and $\boldsymbol{\omega}_C$ if we followed a two-step process, in which we first regress $\boldsymbol{y}$ on $\boldsymbol{t}$

$$\boldsymbol{y} = \boldsymbol{t}\omega_t + \epsilon_1. \tag{5.17}$$

$$\boldsymbol{y} = \boldsymbol{C}\omega_C + \epsilon_1. \tag{5.18}$$

This is holds only true, if and only if $\boldsymbol{t} \perp \boldsymbol{C}$. Because in this case, $\boldsymbol{t}$ and $\boldsymbol{C}$ are already orthorgonal to each other. They already fulfill the requirements of the FWL and therefore such two-step process would yield mathematically equivalent regression coefficients $\boldsymbol{\omega}$ to the joint estimation in equation (5.16). Put in terms of the conditional expectations, given linearity, $\mathbb{E}[\boldsymbol{y}|\boldsymbol{t},\boldsymbol{C}] = \mathbb{E}[\boldsymbol{y}|\boldsymbol{t}] + \mathbb{E}[\boldsymbol{y}|\boldsymbol{C}]$, since $\boldsymbol{t}$ and $\boldsymbol{C}$ are uncorrelated and therefore $\boldsymbol{C}$ is not an actual confounder under the linear CEF setup.

#### 5.3.3.3 $\mathbb{E}[\boldsymbol{y}|\boldsymbol{t},\boldsymbol{C}]$, where $\boldsymbol{t} \not\perp \boldsymbol{C}$, no $\boldsymbol{Z}$

In this case, $\boldsymbol{t} \not\perp \boldsymbol{C}$. We now have $\mathbb{E}[\boldsymbol{y}|\boldsymbol{t},\boldsymbol{C}] \neq \mathbb{E}[\boldsymbol{y}|\boldsymbol{t}]$ in the linear CEF setup, since $\boldsymbol{C}$ is a confounder. However, according to the FWL, we can still conduct separate stage regressions and obtain mathematically equivalent regression coefficients $\boldsymbol{\omega}$ if we residualize outcomes and regressors on all regressors that are not part of the partial regression. We can estimate

$$\boldsymbol{M}_C\boldsymbol{y} = \boldsymbol{M}_C\boldsymbol{t}\widehat{\boldsymbol{\omega}}_t + \hat{\boldsymbol{\epsilon}}_{\mathbf{1}} \tag{5.19}$$

and

$$\boldsymbol{M}_t\boldsymbol{y} = \boldsymbol{M}_t\boldsymbol{C}\widehat{\boldsymbol{\omega}}_{\boldsymbol{C}} + \hat{\boldsymbol{\epsilon}}_{\mathbf{2}} \tag{5.20}$$

and the obtained estimates $\widehat{\omega}_t$ and $\widehat{\boldsymbol{\omega}}_{\boldsymbol{C}}$ will be equivalent to those obtained from the joint estimation.

#### 5.3.3.4 $\mathbb{E}[\boldsymbol{y}|\boldsymbol{t},\boldsymbol{C},\boldsymbol{Z}]$, where $\boldsymbol{t} \not\perp \boldsymbol{C}$,$\boldsymbol{Z}$

We now consider the case where part (or all) of our confounders are text or where text is a proxy for otherwise unobserved confounders. The joint estimation would be

$$\boldsymbol{y} = \boldsymbol{t}\widehat{\boldsymbol{\omega}}_t + \boldsymbol{C}\widehat{\boldsymbol{\omega}}_{\boldsymbol{C}} + \boldsymbol{Z}\widehat{\boldsymbol{\omega}}_{\boldsymbol{Z}} + \hat{\boldsymbol{\epsilon}} \tag{5.21}$$

where $\boldsymbol{Z}$ itself is obtained through supervised learning via text representation function

$$\boldsymbol{Z} = g(\boldsymbol{W},\boldsymbol{y},\boldsymbol{t},\boldsymbol{C};\Theta).$$

We therefore cannot decompose this joint estimation into separate parts. As long as the text features $\boldsymbol{Z}$ are correlated with the outcome and the other covariates, we would need to apply the orthogonalization via the respective $\boldsymbol{M}$ matrices for each partical regression. Since $\boldsymbol{Z}$ needs to be estimated itself (it is 'estimated data'), we cannot residualize on $\boldsymbol{Z}$ though. Nor can $\boldsymbol{Z}$ be residualized on the other covariates. A separate-stage approach will therefore lead to biased estimates of $\boldsymbol{\omega}$.

### 5.3.4 Supervised Topic Representations

Topic models are a popular choice of text representation in causal inference settings (Keith et al., 2020) and in modelling with text as data in social sciences in general (Gentzkow et al., 2019). Veitch et al. (2020) are some of the first to assess both topic and language model representations for causal inference tasks. We focus on the topic representation approach for function $g()$ in our joint modelling strategy, as topic models have been already well established in economics and finance, which are the key application domains of our research. In particular, topic models have proved to work well even in setups with relatively *small* datasets (that is, hundreds to thousands of data points versus millions of data points), which is a common dataset size in macroeconomics and finance. However, our the FWL requirements hold independent of the choice of text representations. We encourage future research to extend our work to different text representation approaches.

**BTR:** We create BTR, a fully Bayesian supervised topic model that can handle numeric metadata as regression features and labels. Its generative process builds on LDA-based models in the spirit of Blei and McAuliffe (2008). Given our focus on causal interpretation, we opt for a Gibbs sampling implementation. This provides statistical guarantees of providing asymptotically exact samples of the target density while (neural) variational inference does not (Robert and Casella, 2013). Blei et al. (2017) point out that MCMC methods are preferable over variational inference when the aim of the task is to obtain asymptotically precise estimates. On the other hand, Gibbs sampling usually comes at a higher computational cost than variational inference alternatives.

**rSCHOLAR:** SCHOLAR (Card et al., 2018) is a supervised topic model that generalises both sLDA (Blei and McAuliffe, 2008) as it allows for predicting labels, and SAGE (Eisenstein et al., 2011) which handles jointly modelling covariates via 'factorising' its topic-word distributions ($\beta$) into deviations from the background log-frequency of words and deviations based on covariates. SCHOLAR is solved via neural variational inference (Kingma and Welling, 2014; Rezende et al., 2014). However, it was not primarily designed for causal inference. We extend SCHOLAR with a linear regression layer (rSCHOLAR) to allow direct comparison with BTR. That is, its downstream layer is $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\omega}$, where $\boldsymbol{A} = [\boldsymbol{t}, \boldsymbol{C}, \boldsymbol{\theta}]$ is the design matrix in which $\boldsymbol{\theta}$ represents the estimated document-topic mixtures. $\boldsymbol{\omega}$ represents the regression weight vector.

This regression layer is jointly optimized with the main SCHOLAR model via backpropagation using ADAM (Kingma and Ba, 2015), replacing the original downstream cross-entropy loss with mean squared error loss.

Other recent supervised topic models that can handle covariates are for example STM (Roberts et al., 2016) and DOLDA (Magnusson et al., 2020). DOLDA was not designed for regression nor for causal inference setups. Topics models in the spirit of STM incorporate document metadata, but in order to better predict the content of documents rather than to predict an outcome. Many approaches on supervised topic models for regression have been suggested over the years. Blei and McAuliffe (2008) optimize their sLDA model with respect to the joint likelihood of the document data and the response variable using VI. MedLDA (Zhu et al., 2012) optimizes with respect to the maximum margin principle, Spectral-sLDA (Wang and Zhu, 2014) proposes a spectral decomposition algorithm, and BPsLDA (Chen et al., 2015) uses backward propagation over a deep neural network. Since BPsLDA reports to outperform sLDA, MedLDA and several other models, we include it in our benchmark list for two-stage models. We include a Gibbs sampled sLDA to have a two-stage model in the benchmark list that is conceptually very similar to BTR in the generative topic modelling part. Unsupervised LDA (Blei et al., 2003; Griffiths and Steyvers, 2004) and a neural topic model counterpart GSM (Miao et al., 2017) are also added for comparison.

## 5.4 Bayesian Topic Regression Model

### 5.4.1 Regression Model

We take a Bayesian approach and jointly estimate $f()$ and $g()$ to solve equation (5.2). To simplify notation, encompass numeric features of treatment $\boldsymbol{t}$ and covariates $\boldsymbol{C}$ in data matrix $\boldsymbol{X} \in \mathbb{R}^{D\times(1+\dim_C)}$. All estimated topic features are represented via $\bar{\boldsymbol{Z}} \in \mathbb{R}^{D\times K}$, where $K$ is the number of topics. Finally, $\boldsymbol{y} \in \mathbb{R}^{D\times 1}$ is the outcome vector. Define $\boldsymbol{A} = [\bar{\boldsymbol{Z}}, \boldsymbol{X}]$ as the overall regression design matrix containing all features (optionally including interaction terms between topics and numeric features). With our fully Bayesian approach, we aim to better capture feature correlations and model uncertainties. In particular, information from the numeric features (labels, treatment, and controls) directly informs the topic assignment process as well as the regression. This counters bias in the treatment effect estimation, following the spirit of

'causally sufficient' text representations (Veitch et al., 2020). Following the previous section, we outline the case for $f()$ being linear. Our framework could however be extended to non-linear $f()$. Assuming Gaussian iid errors $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{I})$, the model's regression equation is then $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{\omega} + \boldsymbol{\epsilon}$, such that

$$p(\boldsymbol{y}|\boldsymbol{A}, \boldsymbol{\omega}, \sigma^2) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{A}\boldsymbol{\omega}, \sigma^2 \boldsymbol{I}). \tag{5.22}$$

The likelihood with respect to outcome $\boldsymbol{y}$ is then

$$p(\boldsymbol{y}|\boldsymbol{A}, \boldsymbol{\omega}, \sigma^2) = \prod_{d=1}^{D} \mathcal{N}(y_d|\boldsymbol{a}_d\boldsymbol{\omega}, \sigma^2 \boldsymbol{I}), \tag{5.23}$$

where $\boldsymbol{a}_d$ is the $d$th row of design matrix $\boldsymbol{A}$. We model our prior beliefs about parameter vector $\boldsymbol{\omega}$ by a Gaussian density

$$p(\boldsymbol{\omega}) = \mathcal{N}(\boldsymbol{\omega}|\boldsymbol{m}_0, \boldsymbol{S}_0) \tag{5.24}$$

where mean $\boldsymbol{m}_0$ and covariance matrix $\boldsymbol{S}_0$ are hyperparameters. Following **?**, we place an Inverse-Gamma prior on the conditional variance estimate $\sigma^2$ with shape and scale hyperparameters $a_0$ and $b_0$

$$p(\sigma^2) = \mathcal{IG}(\sigma^2|a_0, b_0). \tag{5.25}$$

Placing priors on all our regression parameters allows us to conduct full Bayesian inference, which not only naturally counteracts parameter over-fitting but also provides us with well-defined posterior distributions over $\boldsymbol{\omega}$ and $\sigma^2$ as well as a predictive distribution of our response variable.

Due to the conjugacy of the Normal-Inverse-Gamma prior, the regression parameters' posterior distribution has a known Normal-Inverse-Gamma distribution (Stuart et al., 1994)

$$\begin{aligned} p(\boldsymbol{\omega}, \sigma^2|\boldsymbol{y}, \boldsymbol{A}) &\propto p(\boldsymbol{\omega}|\sigma^2, \boldsymbol{y}, \boldsymbol{A}) p(\sigma^2 \mid \boldsymbol{y}, \boldsymbol{A}) \\ &= \mathcal{N}\left(\boldsymbol{\omega}|\boldsymbol{m}_n, \sigma^2 \boldsymbol{S}_n^{-1}\right) \mathcal{IG}\left(\sigma^2|a_n, b_n\right). \end{aligned} \tag{5.26}$$

$\boldsymbol{m}_n$, $\boldsymbol{S}_n$, $a_n$, $b_n$ follow standard updating equations for a Bayesian linear regression. Due to the conjugacy of the Normal-Inverse-Gamma prior, the posterior distribution of the regression parameters conditional on $\boldsymbol{A}$ has a known Normal-Inverse-Gamma distribution:

$$p(\boldsymbol{\omega}, \sigma^2|\boldsymbol{y}, \boldsymbol{A}) \propto p(\boldsymbol{\omega}|\sigma^2, \boldsymbol{y}, \boldsymbol{A}) p(\sigma^2 \mid \boldsymbol{y}, \boldsymbol{A}) = \mathcal{N}\left(\boldsymbol{\omega}|\boldsymbol{m}_n, \sigma^2 \boldsymbol{S}_n^{-1}\right) \mathcal{IG}\left(\sigma^2|a_n, b_n\right) \tag{5.27}$$

where $\boldsymbol{m}_n$, $\boldsymbol{S}_n$, $a_n$ and $b_n$ follow standard updating equations for a Bayesian Linear Regression

(Bishop 2006)

$$\boldsymbol{m}_n = (\boldsymbol{A}^\intercal \boldsymbol{A} + \boldsymbol{S}_0)^{-1}(\boldsymbol{S}_0 \boldsymbol{m}_0 + \boldsymbol{A}^\intercal \boldsymbol{y}) \tag{5.28}$$

$$\boldsymbol{S}_n = (\boldsymbol{A}^\intercal \boldsymbol{A} + \boldsymbol{S}_0) \tag{5.29}$$

$$a_n = a_0 + N/2 \tag{5.30}$$

$$b_n = b_0 + (\boldsymbol{y}^\intercal \boldsymbol{y} + \boldsymbol{m}_0^\intercal \boldsymbol{S}_0 \boldsymbol{m}_0 - \boldsymbol{m}_n^\intercal \boldsymbol{S}_n \boldsymbol{m}_n)/2. \tag{5.31}$$

### 5.4.2 Topic Model

The estimated topic features $\bar{\boldsymbol{Z}}$, which form part of the design regression matrix $\boldsymbol{A}$, are generated from a supervised model that builds on an LDA-based topic structure (Blei et al., 2003). Figure 5.1 provides a graphical representation of BTR and brings together our topic and regression model.

We have $d$ documents in a corpus of size $D$, a vocabulary of $V$ unique words and $K$ topics. A document has $N_d$ words, so that $w_{d,n}$ denotes the $n$th word in document $d$. The bag-of-words representation of a document is $\boldsymbol{w}_d = [w_{d,1}, \ldots, w_{d,N_d}]$, so that the entire corpus of documents is described by $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_D]$. $z_{d,n}$ is the topic assignment of word $w_{d,n}$, where $\boldsymbol{z}_d$ and $\boldsymbol{Z}$ mirror $\boldsymbol{w}_d$ and $\boldsymbol{W}$ in their dimensionality. Similarly, $\bar{\boldsymbol{z}}_d$ denotes the estimated average topic assignments of the $K$ topics across words in document $d$, such that $\bar{\boldsymbol{Z}} = [\bar{\boldsymbol{z}}_{\boldsymbol{1}}, \ldots, \bar{\boldsymbol{z}}_{\boldsymbol{D}}]^\intercal \in \mathbb{R}^{D \times K}$. $\boldsymbol{\beta} \in \mathbb{R}^{K \times V}$, describes the $K$ topic distributions over the $V$ dimensional vocabulary. $\boldsymbol{\theta} \in \mathbb{R}^{D \times K}$ describes the $K$ topic mixtures for each of the $D$ documents. $\eta \in \mathbb{R}^V$ and $\alpha \in \mathbb{R}^K$ are the respective hyperparameters of the prior for $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. The generative process of our BTR model is then:

1. $\boldsymbol{\omega} \sim \mathcal{N}(\boldsymbol{\omega}|\boldsymbol{m}_0, \boldsymbol{S}_0)$ and $\sigma^2 \sim \mathcal{IG}(\sigma^2|a_0, b_0)$
2. **for** $k = 1, \ldots, K$:
   (a) $\boldsymbol{\beta}_k \sim \text{Dir}(\eta)$
3. **for** $d = 1, \ldots, D$:
   (a) $\boldsymbol{\theta}_d \sim \text{Dir}(\alpha)$
   (b) **for** $n = 1, \ldots, N_d$:

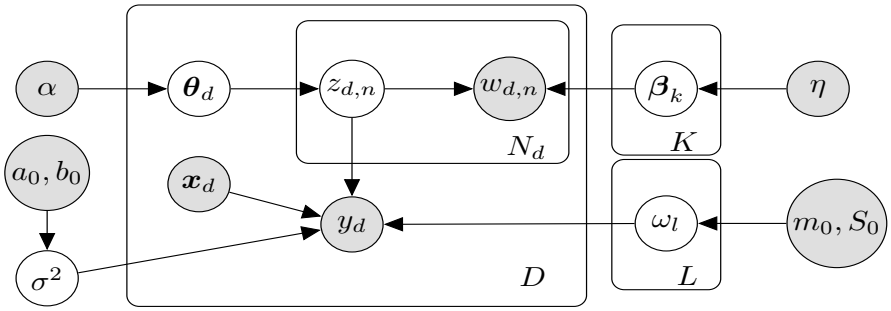
i. topic assignment $z_{d,n} \sim Mult(\boldsymbol{\theta}_d)$

ii. term $w_{d,n} \sim Mult(\boldsymbol{\beta}_{z_{d,n}})$

4. $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{\omega}, \sigma^2\boldsymbol{I})$.

Straightforward extensions also allow multiple documents per observation or observations without documents, as is described in Appendix B.1.

Figure 5.1: Graphical model for BTR.




## 5.5 Estimation

### 5.5.1 Posterior Inference

The objective is to identify the latent topic structure and regression parameters that are most probable to have generated the observed data. We obtain the joint distribution for our graphical model through the product of all nodes conditioned only on their parents, which for our model is

$$\begin{aligned} &p(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2 | \boldsymbol{X}, \alpha, \eta, \boldsymbol{m}_0, \boldsymbol{S}_0, a_0, b_0) = \\ &\prod_{d=1}^{D} p(\boldsymbol{\theta}_d|\alpha) \prod_{k=1}^{K} p(\boldsymbol{\beta}_k|\eta) \prod_{d=1}^{D} \prod_{n=1}^{N_d} p(z_{d,n}|\boldsymbol{\theta}_d) p(w_{d,n}|z_{d,n}, \boldsymbol{\beta}) \\ &\prod_{d=1}^{D} p(y_d|\boldsymbol{x}_d, \boldsymbol{z}_d, \boldsymbol{\omega}, \sigma^2) \prod_{l=1}^{L} p(\boldsymbol{\omega}_l|\boldsymbol{m}_0, \boldsymbol{S}_0) p(\sigma^2|a_0, b_0). \end{aligned} \tag{5.32}$$

The inference task is thus to compute the posterior distribution of the latent variables ($\boldsymbol{Z}$, $\boldsymbol{\theta}$, $\boldsymbol{\beta}$, $\boldsymbol{\omega}$, and $\sigma^2$) given the observed data ($\boldsymbol{y}$, $\boldsymbol{X}$ and $\boldsymbol{W}$) and the priors governed by hyperparameters ($\alpha, \eta$, $\boldsymbol{m}_0, \boldsymbol{S}_0, a_0, b_0$). We will omit hyperparameters for sake of clarity unless explicitly needed for computational steps. The posterior distribution is then

$$p(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{Z}, \boldsymbol{\omega}, \sigma^2 | \boldsymbol{W}, \boldsymbol{y}, \boldsymbol{X}) = \frac{p(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2)}{p(\boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y})}. \tag{5.33}$$

In practice, computing the denominator in equation (5.33), i.e. the evidence, is intractable due to the sheer number of possible latent variable configurations. We use a Gibbs EM algorithm

(Levine and Casella, 2001) set out below, to approximate the posterior. Collapsing out the latent variables $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ (Griffiths and Steyvers, 2004), we only need to identify the sampling distributions for topic assignments $\boldsymbol{Z}$ and regression parameters $\boldsymbol{\omega}$ and $\sigma^2$, conditional on their Markov blankets

$$\begin{aligned} &p(\boldsymbol{Z}, \boldsymbol{\omega}, \sigma^2 | \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}) = \\ &p(\boldsymbol{Z} | \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2) p(\boldsymbol{\omega}, \sigma^2 | \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{y}). \end{aligned} \tag{5.34}$$

Once topic assignments $\boldsymbol{Z}$ are estimated, it is straightforward to recover $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. The expected topic assignments are estimated by Gibbs sampling in the E-step, and the regression parameters are estimated in the M-step.

### 5.5.2 E-Step: Estimate Topic Parameters

In order to sample from the conditional posterior for each $z_{d,n}$ we need to identify the probability of a given word $w_{d,n}$ being assigned to a given topic $k$, conditional on the assignments of all other words (as well as the model's other latent variables and the observed data)

$$p(z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2), \tag{5.35}$$

where $\boldsymbol{Z}_{-(d,n)}$ are the topic assignments of all words apart from $w_{d,n}$. This section defines this distribution, with derivations in Appendix B.2.

By conditional independence properties of the graphical model, we can split this joint posterior into

$$p(\boldsymbol{Z} | \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2) \propto p(\boldsymbol{Z} | \boldsymbol{W}) p(\boldsymbol{y} | \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{\omega}, \sigma^2). \tag{5.36}$$

Topic assignments within one document are independent from topic assignments in all other documents and the sampling equation for $z_{d,n}$ only depends on it's own response variable $y_d$, hence

$$\begin{aligned} &p(z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2) \propto \\ &p(z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}) p(y_d | z_{d,n} = k, \boldsymbol{Z}_{-(d,n)}, \boldsymbol{x}_d, \boldsymbol{\omega}, \sigma^2). \end{aligned} \tag{5.37}$$

The first part of the RHS expression is the sampling distribution of a standard LDA model. Following Griffiths and Steyvers (2004), we can express it in terms of count variables $s$ (topic assignments across a document) and $m$ (assignments of unique words across topics over all

documents).[4]

The second part is the predictive distribution for $y_d$. This is a Gaussian distribution depending on the linear combination $\boldsymbol{\omega}(\boldsymbol{a}_d|z_{d,n}=k)$, where $\boldsymbol{a}_d$ includes the topic proportions $\bar{\boldsymbol{z}}_d$ and $\boldsymbol{x}_d$ variables (and any interaction terms), conditional on $z_{d,n}=k$. We can write this in a convenient form that preserves proportionality with respect to $z_{d,n}$ and depends only on the data and the count variables.

First, we split the $\boldsymbol{X}$ features into those that are interacted, $\boldsymbol{X}_{1,d}$, and those that are not, $\boldsymbol{X}_{2,d}$ such that the generative model for $y_d$ is then

$$y_d \sim \mathcal{N}(\boldsymbol{\omega}_z^\intercal \bar{\boldsymbol{z}}_d + \boldsymbol{\omega}_{zx}^\intercal(\boldsymbol{x}_{1,d} \otimes \bar{\boldsymbol{z}}_d) + \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_{2,d}, \sigma^2), \tag{5.38}$$

where $\otimes$ is the Kronecker product. Define $\tilde{\boldsymbol{\omega}}_{z,d}$ as a length $K$ vector such that

$$\tilde{\omega}_{z,d,k} = \omega_{z,k} + \boldsymbol{\omega}_{zx,k}^\intercal \boldsymbol{x}_{1,d}. \tag{5.39}$$

Noting that $\tilde{\boldsymbol{\omega}}_{z,d}^\intercal \bar{\boldsymbol{z}}_d = \frac{\tilde{\boldsymbol{\omega}}_{z,d}^\intercal}{N_d}(\boldsymbol{s}_{d,-n} + s_{d,n})$, gives us the sampling distribution for $z_{d,n}$ stated in equation (5.37): a multinomial distribution parameterised by

$$\begin{aligned} p(z_{d,n}=k|z_{-(d,n)}, W, X, y, \alpha, \eta, \omega, \sigma^2) \propto (s_{d,k,-n}+\alpha)\times \\ \frac{m_{k,v,-(d,n)}+\eta}{\sum_v m_{k,v,-(d,n)}+V\eta} \exp\left\{\frac{1}{2\sigma^2}\left(\frac{2\tilde{\omega}_{z,d,k}}{N_d}(y_d - \right.\right. \\ \left.\left. \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_{2,d} - \frac{\tilde{\boldsymbol{\omega}}_{z,d}^\intercal}{N_d}\boldsymbol{s}_{d,-n} - \left(\frac{\tilde{\omega}_{z,d,k}}{N_d}\right)^2\right)\right\}. \end{aligned} \tag{5.40}$$

This defines the probability for each $k$ that $z_{d,n}$ is assigned to that topic $k$. These $K$ probabilities define the multinomial distribution from which $z_{d,n}$ is drawn.

### 5.5.3 M-Step: Estimate Regression Parameters

To estimate the regression parameters, we hold the design matrix $\boldsymbol{A} = [\bar{\boldsymbol{Z}}, \boldsymbol{X}]$ fixed. Given the Normal-Inverse-Gamma prior, this is a standard Bayesian linear regression problem and the posterior distribution for which is given in equation (5.26) above. To prevent overfitting to the training sample there is the option to randomly split the training set into separate sub-samples for the E- and M-steps, following a Cross-Validation EM approach (Shinozaki and Ostendorf,

[4]For example, $s_{d,k}$ denotes the total number of words in document $d$ assigned to topic $k$ and $s_{d,k,-n}$ the number of words in document $d$ assigned to topic $k$, except for word $n$. Analogously, $m_{k,v}$ measures the total number of times term $v$ is assigned to topic $k$ across all documents and $m_{k,v,-(d,n)}$ measures the same, but excludes word $n$ in document $d$.

2007). We use the prediction mean squared error from the M-step sample to assess convergence across EM iterations.

### 5.5.4 Implementation

We provide an efficient *Julia* implementation for BTR and a *Python* implementation for rSCHOLAR on Github to allow for reproducibility of the results in the following experiment sections.[5]

## 5.6 Experiment: Synthetic Data

### 5.6.1 Synthetic Data Generation

To illustrate the benefits of our BTR approach, we generate a synthetic dataset of documents which have explanatory power over a response variable, along with an additional numeric covariate that is correlated with both documents and response.

We generate $10,000$ documents of 50 words each, following an LDA generative process, with each document having a distribution over three topics, defined over a vocabulary of 9 unique terms. A numeric feature, $\boldsymbol{x} = [x_1, ..., x_D]^\intercal$, is generated by calculating the document-level frequency of the first word in the vocabulary. As the first topic places a greater weight on the first three terms in the vocabulary, $\boldsymbol{x}$ is positively correlated with $\bar{\boldsymbol{z}}_1$. The response variable $\boldsymbol{y} = [y_1, ..., y_D]$ is generated through a linear combination of the numeric feature $\boldsymbol{x}$ and the average topic assignments $\bar{\boldsymbol{Z}} = \{\bar{\boldsymbol{z}}_1, \bar{\boldsymbol{z}}_2, \bar{\boldsymbol{z}}_3\}$,

$$\boldsymbol{y} = -\bar{\boldsymbol{z}}_1 + \boldsymbol{x} + \boldsymbol{\epsilon}. \tag{5.41}$$

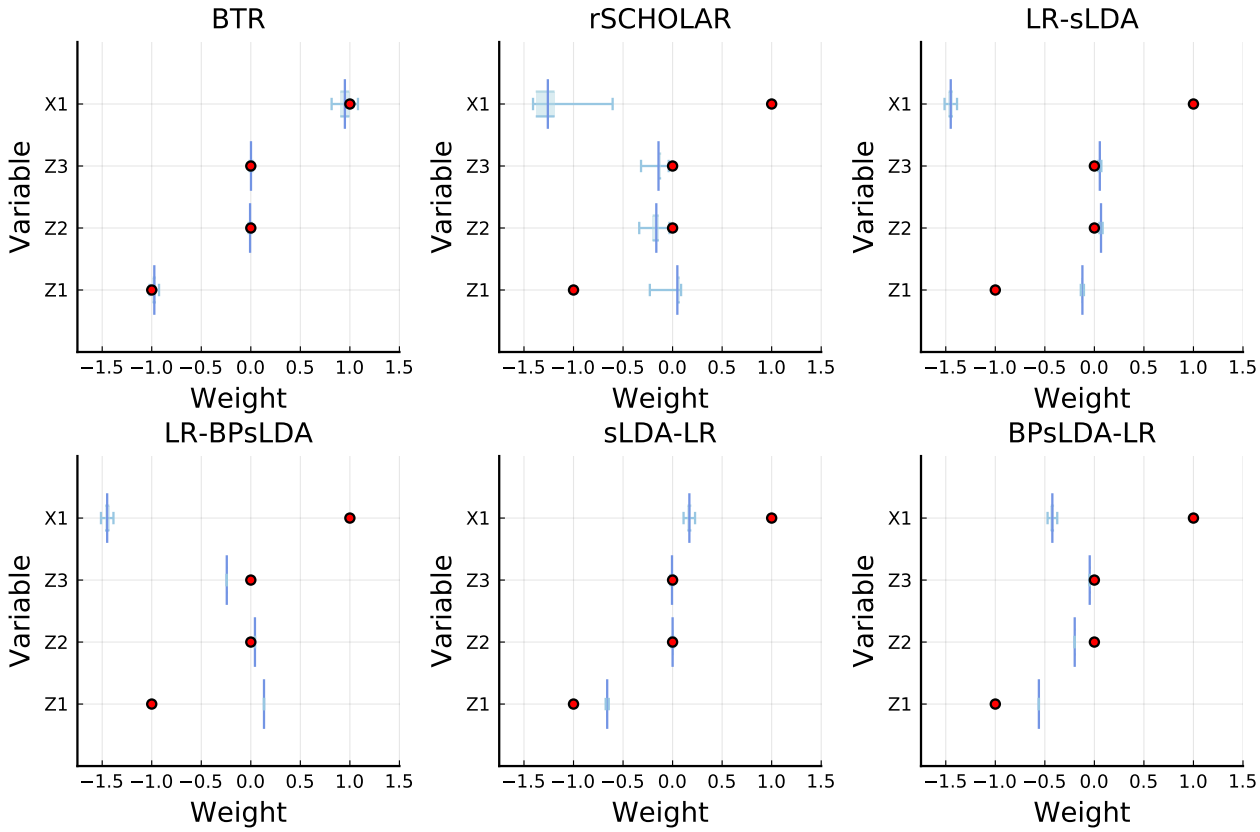
where $\boldsymbol{\epsilon}$ is an iid Gaussian white noise term. The regression model to recover the ground truth is then

$$\boldsymbol{y} = \omega_1 \bar{\boldsymbol{z}}_1 + \omega_2 \bar{\boldsymbol{z}}_2 + \omega_3 \bar{\boldsymbol{z}}_3 + \omega_4 \boldsymbol{x}_d + \boldsymbol{\epsilon}. \tag{5.42}$$

The *true* regression weights are thus $\boldsymbol{\omega}^* = [-1, 0, 0, 1]$. In accordance with the FWL theorem, we cannot recover the true coefficients with a two-stage estimation process.

[5] BTR: github.com/julianashwin/BTR.jl
rSCHOLAR: github.com/MaximilianAhrens/scholar4regression

Figure 5.2: Comparing recovery of true regression weights across different topic models




*For each panel, the true regression weights are shown as red points and the estimated* 95% *posterior credible (or bootstrap, depending on model) interval in blue. Only BTR contains the true weights within the estimated intervals.*

### 5.6.2 Synthetic Data Results

We compare the ground truth of the synthetic data generating process against: (1) **BTR:** our Bayesian model, estimated via Gibbs sampling. (2) **rSCHOLAR**: the regression extension of SCHOLAR, estimated via neural VI. (3) **LR-sLDA:** first linearly regress $\boldsymbol{y}$ on $\boldsymbol{x}$, then use the residual of that regression as the response in an sLDA model, estimated via Gibbs sampling. (4) **sLDA-LR:** First sLDA, then linear regression. (5) **BPsLDA-LR** and (6) **LR-BPsLDA:** replace sLDA with BPsLDA, which is sLDA estimated via the backpropagation approach of Chen et al. (2015).

Figure 5.2 shows the true and estimated regression weights for each of the six models. LR-sLDA and sLDA-LR estimate inaccurate regression weights for both the text and numeric features, as do the BPsLDA variants. Similarly, rSCHOLAR fails to recover the ground truth. However, BTR estimates tight posterior distributions around to the true parameter values. The positive correlation between $\boldsymbol{z}_1$ and $\boldsymbol{x}$ makes a joint estimation approach crucial

for recovering the true parameters. Standard supervised topic models estimate the regression parameters for the numeric features separately from the topic proportions and their associated regression parameters, violating the FWL theorem as outlined in Section 5.3.2. A key difference between rSCHOLAR and BTR lies in their posterior estimation techniques (neural VI vs Gibbs). rSCHOLAR's approach seems to have a similarly detrimental effect as the two-stage approaches. We suspect further research into (neural) VI assumptions and their effect on causal inference with text could be fruitful.

## 5.7 Experiment: Semi-Synthetic Data

### 5.7.1 Semi-Synthetic Data Generation

We further benchmark the models' abilities to recover the ground truth on two semi-synthetic datasets. In those datasets, we still have access to the ground truth (GT) as we either synthetically create or directly observe the correlations between treatment, confounders and outcome. However, the text and some numeric metadata that we use is empirical. We use customer review data from **(i) Booking.com**[6] and **(ii) Yelp**[7], and analyse two different 'mock' research questions. For both datasets, we randomly sample $50,000$ observations and select 75% in Yelp, 80% in Booking for training.[8]

**Booking:** *Do people give more critical ratings ($y_i$) to hotels that have high historic ratings ($av\_score_i$), once controlling for review texts?*

$$GT_B: \quad y_i = -\text{hotel\_av}_i + 5\text{prop\_pos}_i \tag{5.43}$$

where $\text{prop\_pos}_i$ is the proportion of positive words in a review. The textual effect is estimated via topic modelling in our experiment. The treatment in question is the average historic customer rating, being modelled as continuous.

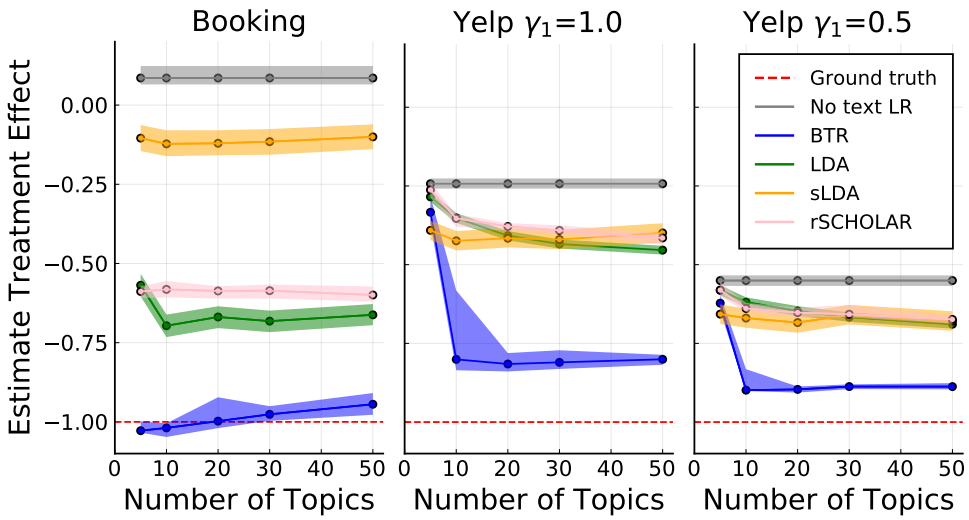
**Yelp:** *Do people from the US ($US_i$=1) give different Yelp ratings ($y_i$) than customers from*

[6] Available at kaggle.com/jiashenliu

[7] Available at yelp.com/dataset, Toronto subsample

[8] Appendix B.5 for full data summary statistics. Data samples used for experiments available via: github.com/MaximilianAhrens/data

Figure 5.3: Estimated treatment effects on semi-synthetic data




*Booking (left panel), Yelp (middle and right panel). Intervals are either 95% credible interval of posterior distribution, or based on 20 run bootstrap, depending on model.*

*Canada ($US_i$=0), controlling for average restaurant review ($stars\_av\_b_i$) and the review text?*

$$GT_Y : y_i = -\text{US}_i + \text{stars\_av\_b}_i + sent_i. \tag{5.44}$$

To create the binary treatment variable $\text{US}_i$, we compute each review's sentiment score ($sent_i$) using the Harvard Inquirer. This treatment effect is correlated with the text as

$$\Pr(US_i = 1) = \frac{\exp(\gamma_1 sent_i)}{1 + \exp(\gamma_1 sent_i)}, \tag{5.45}$$

where $\gamma_1$ controls the correlation between text and treatment.[9]

### 5.7.2 Semi-Synthetic Data Results

On both semi-synthetic datasets and across all benchmarked models, BTR estimates the regression weights that are the closest to the ground truth. This consistently holds true across all tested numbers of topics $K$ (see Figure D.6). For Yelp, we also vary the correlation strength between treatment and confounder. The middle panel in Figure D.6 shows the estimation results with a very high correlation between confounder and treatment ($\gamma_1 = 1$). The RHS panel shows the results when this correlation is lower ($\gamma_1 = 0.5$). As expected, a higher correlation between confounder and treatment increases the bias as outlined in Section 5.3.2. If the correlation between confounder and treatment is zero, a two-stage estimation approach no longer violates

[9]When $\gamma_1 = 1$, correlation between $US_i$ and $sent_i$ is 0.23. For $\gamma_1 = 0.5$ it is 0.39.

FWL and all models manage to estimate the ground truth (see Appendix B.4). Since the topic modelling approach is an approximation to capture the true effect of the text and its correlation with the metadata - and since this approximation is not perfect - some bias may remain. Overall, BTR gets substantially closer to the ground truth than any other model.

## 5.8 Experiment: Real-World Data

The joint supervised estimation approach using text and non-text features, not only counteracts bias in causal settings. It also improves prediction performance. We use the real-world datasets of Booking and Yelp for our benchmarking. For both datasets, we predict customer ratings (response) for a business or hotel given customer reviews (text features) and business and customer metadata (numeric features).[10]

### 5.8.1 Benchmarks

We add the following models to the benchmark list from the previous section:[11] **LDA+LR** (Griffiths and Steyvers, 2004) and **GSM+LR** (Miao et al., 2017) unsupervised Gibbs sampling and neural VI based topic models. **LR+rSCHOLAR**: the two-step equivalent for rSCHOLAR, estimating covariate regression weights in a separate step from the supervised topic model.

An alternative to topic based models are word-embedding based neural networks. We use (7) **LR+aRNN**: a bidirectional RNN with attention (Bahdanau et al., 2015). Since the model does not allow for non-text features, we use the regression residuals of the linear regression as the target. And (8) **LR+TAM**: a bidirectional RNN using global topic vector to enhance its attention heads (Wang and Yang, 2020) - same target as in LR+aRNN. [12]

### 5.8.2 Prediction and Perplexity Results

We evaluated all topic models on a range from 10 to 100 topics, with results for 50 and 100 in Table 5.2.[13] Hyperparameters of benchmark models that have no direct equivalent in our

[10] full specifications for each case are given in Appendix B.5

[11] We also tested sLDA+LR and a pure sLDA, which performed consistently worse, see Appendix B.6.1

[12] Wang and Yang (2020) use 100-dimensional word embeddings in their default setup for TAM and pre-train those on the dataset. We follow this approach. RNN and TAM results were very robust to changes in the hidden layer size in these setups, we use a layer size of 64. Full details of all model parametrisations are provided in Appendix B.6.2.

[13] Hyperparameters of displayed results: $\alpha = 0.5$, $\eta = 0.01$

model were set as suggested in the pertaining papers. We find that our results are robust across a wide range of hyperparameters (extensive robustness checks in Appendix B.6).

Table 5.1: Results: Booking

| *Dataset* | Booking | |
|---|---|---|
| $K$ | 50 | 100 |
| $pR^2$ (higher is better) | | |
| OLS | 0.315 | |
| aRNN | 0.479 (0.007) | |
| LR+ TAM | 0.479 (0.014) | 0.487 (0.014) |
| LDA+LR | 0.426 (0.003) | 0.437 (0.002) |
| GSM+LR | 0.386 (0.004) | 0.395 (0.005) |
| LR+sLDA | 0.432 (0.002) | 0.438 (0.004) |
| LR+BPsLDA | 0.419 (0.009) | 0.455 (0.001) |
| LR+rSCHOLAR | 0.469 (0.002) | 0.465 (0.002) |
| **rSCHOLAR** | **0.494** (0.004) | **0.489** (0.003) |
| **BTR** | 0.454 (0.003) | 0.460 (0.002) |
| Perplexity (lower is better) | | |
| LR+TAM | 521 (2) | 522 (2) |
| LDA+LR | 454 (1) | 432 (1) |
| GSM+LR | **369** (8) | **348** (5) |
| LR+sLDA | 436 (2) | 411 (1) |
| LR+rSCHOLAR | 441 (20) | 458 (11) |
| **rSCHOLAR** | 466 (19) | 464 (9) |
| **BTR** | 437 (1) | 412 (1) |

*Mean $pR^2$ and perplexity, standard deviation in brackets. 20 model runs. Best model **bold**.*

Table 5.2: Results: Yelp

| *Dataset* | Yelp | |
|---|---|---|
| $K$ | 50 | 100 |
| $pR^2$ (higher is better) | | |
| OLS | 0.451 | |
| aRNN | 0.582 (0.008) | |
| LR+ TAM | 0.585 (0.012) | 0.587 (0.008) |
| LDA+LR | 0.586 (0.006) | 0.606 (0.007) |
| GSM+LR | 0.495 (0.004) | 0.517 (0.007) |
| LR+sLDA | 0.571 (0.002) | 0.574 (0.001) |
| LR+BPsLDA | 0.603 (0.002) | 0.609 (0.001) |
| LR+rSCHOLAR | 0.550 (0.034) | 0.557 (0.027) |
| **rSCHOLAR** | 0.571 (0.01) | 0.581 (0.009) |
| **BTR** | **0.630** (0.001) | **0.633** (0.001) |
| Perplexity (lower is better) | | |
| LR+TAM | 1661 (7) | 1655 (7) |
| LDA+LR | 1306 (4) | 1196 (2) |
| GSM+LR | 1431 (34) | 1387 (14) |
| LR+sLDA | 1294 (5) | 1174 (3) |
| LR+rSCHOLAR | 1515 (34) | 1516 (30) |
| **rSCHOLAR** | 1491 (9) | 1490 (9) |
| **BTR** | **1291 (5)** | **1165** (3) |

*Mean $pR^2$ and perplexity, standard deviation in brackets. 20 model runs. Best model **bold**.*

We assess the models' predictive performance based on predictive $R^2$ ($pR^2 = 1 - \frac{\text{MSE}}{var(y)}$). The upper part of Table 5.2 shows that BTR achieves the best $pR^2$ in the Yelp dataset and and very competitive results in the Booking dataset, where our rSCHOLAR extension outperforms all other models. Even the non-linear neural network models aRNN and TAM cannot achieve better results. Importantly, rSCHOLAR and BTR perform substantially better than their counterparts that do not jointly estimate the influence of covariates (LR+rSCHOLAR and LR+sLDA).

To assess document modelling performance, we report the test set perplexity score for all models that allow this (Table 5.2, bottom panel) . Perplexity is defined as $\exp\left\{-\frac{\sum_{d=1}^{D}\log p(\boldsymbol{w}_d|\boldsymbol{\theta},\boldsymbol{\beta})}{\sum_{d=1}^{D}N_d}\right\}$. The joint approach of both rSCHOLAR and BTR does not come at the cost of increased perplexity. If anything, the supervised learning approach using labels and covariates even improves document modelling performance when compared against its unsupervised counterpart (BTR vs LDA).

Assessing the interpretability of topic models is ultimately a subjective exercise. In Appendix

B.6.4 we show topics associated with the most positive and negative regression weights, for each dataset. Overall, the identified topics and the sign of the associated weights seem interpretable and intuitive.

## 5.9 Discussion

In this paper, we introduced BTR, a Bayesian topic regression framework that incorporates both numeric and text data for modelling a response variable, jointly estimating all model parameters. Motivated by the FWL theorem, this approach is designed to avoid potential bias in the regression weights, and can provide a sound regression framework for statistical and causal inference when one needs to control for both numeric and text based confounders in observational data. We demonstrate that our model recovers the ground truth with lower bias than any other benchmark model on synthetic and semi-synthetic datasets. Experiments on real-world data show that a joint and supervised learning strategy also yields superior prediction performance compared to 'two-stage' strategies, even competing with deep neural networks. Finally, a word on computational efficiency. For causal inference, our Gibbs sampling based BTR approach yields the most unbiased coefficient estimates of all considered models. We therefore suggest opting for BTR if the identification of unbiased coefficient estimates is of high importance to the research task. For pure predictions exercises, we observed that the models following a joint and supervised learning strategy (BTR and rSCHOLAR) showed the best performances. BTR performed best on the Yelp dataset, rSCHOLAR on the Booking dataset. As rSCHOLAR comes at much lower computational costs compared to BTR, it might be preferable to use rSCHOLAR for prediction tasks on large datasets.

# 6

# Central Bank Communication and High-Frequency Market Responses

In this chapter, we introduce our empirical identification framework for monetary news shocks, based on our multimodal NLP modelling approach.

**Authors**[1]**:** Maximillian Ahrens[2], Deniz Erdemlioglu, Michael McMahon, Christopher J. Neely, Xiye Yang

**Publication venue:** Accepted for *ECONDAT 2023 Spring Meeting: Economics with Nontraditional Data and Analytical Tools - May 2023*, and *2023 Annual Meeting of the Central Bank Research Association*; accepted for publication in the *Journal of Econometrics* (December 2024)

**Note:** The version published in the Journal of Econometrics is an updated and improved version that differs in parts from the version in this thesis. In particular, the analyses on economic regimes have been revised and removed from the ultimately published paper. Other, new research findings have been added in the published version instead. The change in structure is partly because after carrying out additional tests, we discovered further research angles on regime-dependency that we wanted to fully investigate before writing our final judgment on that matter. As we haven't had an all conclusive evidence base by the time of publication, we left the regime-dependence largely out of the final version – leaving it as a promising research questions for future papers. We therefore encourage the reader of this thesis to see the statements and sections on regime-dependence in this chapter as first preliminary explorations into this matter rather than final and conclusive results. We leave it for future research to conclusively refine,

[1]The views expressed are those of the individual authors and do not necessarily reflect official positions of the Federal Reserve Bank of St. Louis, the Federal Reserve System, or the Board of Governors.

[2]main author

prove, and disprove findings in this area.

## 6.1 Abstract

Researchers have carefully studied post-meeting central bank communication and have found that it often moves markets, but they have paid less attention to the more frequent central bankers' speeches. We create a novel dataset of US Federal Reserve speeches and use supervised multimodal natural language processing methods to identify how monetary policy news affect financial volatility and tail risk through implied changes in forecasts of GDP, inflation, and unemployment. We find that news in central bankers' speeches can help explain volatility and tail risk in both equity and bond markets. Our results challenge the conventional view that central bank communication primarily resolves uncertainty.

## 6.2 Introduction

A large branch of monetary policy research seeks to explain how central bank communication (CBC) steers market dynamics and expectations (Blinder, 2018). Theory suggests that if central bank announcements and speeches convey information on economic and monetary conditions, market participants will update their beliefs as reflected in their portfolio choices. Central bank communication can thus contribute to revaluing assets and stabilizing market conditions by reducing uncertainty (Bernanke et al., 2005). Empirical research largely corroborates this theoretical prediction and establishes a consensus that central bank communication influences asset prices through its effects on market participants' expectations about economic outlook and policy decisions (Bernanke and Kuttner, 2005; Ramey, 2016). Monetary policy communication also appears to influence investors' risk aversion and hence the risk premium (Hanson and Stein, 2015; Cieslak and Schrimpf, 2019; Swanson, 2021).

Despite these findings, there are still at least two unresolved issues: (i) how to identify monetary policy news in central bank communication, and (ii) how to identify effects of such news on market uncertainty, i.e., volatility and tail risk. Official central bank announcement dates, such as those of FOMC announcements, occur rather infrequently (every 6-8 weeks). However, policy makers and researchers have suggested that markets continually revise their understanding of central bank information as policy makers give speeches (Neuhierl and Weber, 2019). Although recent developments in natural language processing (NLP) have allowed

economists to analyse text with machine learning methods (see e.g., Bholat et al., 2015; Hansen et al., 2018; Ahrens and McMahon, 2021), researchers have paid only limited attention to speeches so far[3], partly because their content is difficult to quantify and the field still lacks easily accessible datasets of central bank speeches.

In this paper, we develop a novel multimodal NLP method to identify macroeconomic news in central bank speeches and we assess their impact on market volatility and tail risk. To the best of our knowledge, we are the first to do so. Some earlier research has focused on how central bank communication affects volatility in financial markets (see e.g., Bekaert et al., 2013; Cieslak and Schrimpf, 2019; Ehrmann and Talmi, 2020; Gómez-Cram and Grotteria, 2022), while only Hattori et al. (2016) has studied tail risk.[4] Moreover, there is an extensive literature that studies the effects of central bank communication about the economic outlook on asset price surprises. Signals about the economic situation can have a multitude of different effects. The classic channel as emphasised in, for example, Romer and Romer (2000) and Nakamura and Steinsson (2018), is an information effect. The central bank, either explicitly or implicitly through its policy decision, releases superior information about the economy and this information is then incorporated in updated private sector forecasts. An alternative channel is one in which the central bank's information is not considered superior; releasing an alternative assessment of the state of the economy, that the market do not believe, could heighten concerns about the possibility of a monetary policy mistake which would make the economy more volatile (Caballero and Simsek, 2022; Cieslak and McMahon, 2023). The central bank may communicate, as part of its outlook, their view of uncertainty which can influence private views about uncertainty (Hansen et al., 2019). Finally, a cacophony of economic assessments, even if just reflecting different views on the outlook for the economy, might itself signal greater uncertainty surrounding the outlook which can increase the uncertainty of market participants about the economic and the policy outlook (Ahrens and McMahon, 2021).

Our methodological framework has two parts. First, we use machine learning methods from

[3]Recently, Neuhierl and Weber (2019) have investigated the tone of speeches by central bank chairs and vice-chairs while Petropoulos and Siakoulis (2021) use a mixture of machine learning and dictionary methods to calculate sentiment indices from central bank speeches. The latter authors argue that this sentiment predicts financial turmoil. Swanson (2023) highlights the importance of Fed Chair speeches using an event-study surprise decomposition, and Cieslak and McMahon (2023) focus on the communication of Fed stance and its effects on the risk premium.

[4]We focus on measuring market uncertainty rather than uncertainty about monetary policy (see e.g., Bauer et al., 2022; Husted et al., 2020; Ozdagli and Velikov, 2020; Tillmann, 2020), or uncertainty of monetary policymakers Cieslak et al. (2023).

the field of multimodal natural language processing to infer implied macroeconomic forecast revisions from Fed officials' public speeches. Our training dataset consists of Greenbook texts and their respective forecasts, which allows us to learn a mapping from central bank language to central bank forecasts (see Ahrens and McMahon, 2021). In our test dataset, we then apply the learned mapping to central bank speeches to infer how news signals in speeches can predict revisions of public macroeconomic forecasts. Second, we investigate the high-frequency (intradaily) responses of market volatility and tail risk to speech-implied revisions in CPI, GDP, and unemployment outlooks.[5]

Our paper contributes to the literature in several ways. Most importantly, we show that central bankers' speeches have a statistically significant impact on volatility and tail risk in financial markets. In order to show this, we develop a new, multimodal methodological framework for identifying monetary policy news about GDP growth, CPI, and unemployment outlooks. We compare and contrast the performance of an extensive array of modern machine learning methods for multimodal NLP on our empirical datasets of Greenbook texts and forecasts as well as on FOMC members' speeches. We show that our speech-implied forecast revisions predict future changes in Survey of Professional Forecasters (SPF) forecasts substantially better than models that use purely numeric data and ignore the textual content of the speeches. It is these speech-implied macroeconomic news signals that explain a sizeable part of realized volatility and tail risk in financial markets. In order to contribute to future examinations of Federal Reserve speeches, we make our comprehensive dataset on Federal Reserve speeches accessible to other researchers.

The remainder of the paper is organized as follows. In the next section, we review the related literature. Section 6.4 describes the data and section 6.5 introduces our methodological framework. In section 6.6 and 6.7, we present the empirical results pertaining to our analyses of speech-implied news and high-frequency market responses. Section 6.8 concludes the paper.

[5]High-frequency market analysis is common in monetary research; see, for example, Gurkaynak et al. (2005); Gertler and Karadi (2015); Nakamura and Steinsson (2018); Jarociński and Karadi (2020) and Miranda-Agrippino and Ricco (2021).

## 6.3 Related Literature

### Central Bank Communication Effects on Volatility and Tail Risk

Our paper is most closely related to studies of the high-frequency effects of CBC on market uncertainty and volatility. Cieslak and Schrimpf (2019) study the high-frequency effects of the non-monetary news component of communication on volatility. Leombroni et al. (2021) explore how CBC influences credit risk premia through high-frequency changes in yield curve. Ehrmann and Talmi (2020) measure textual differences between central bank announcements and find that higher levels of textual similarity to the previous announcement statement are usually associated with lower market volatility after the announcement date. Relying on a one-day event window, Hansen et al. (2019) analyse the Bank of England's Inflation Reports via topic modelling and find that communication of uncertainty plays an important role in shaping long-run interest rates. Bekaert et al. (2013) find evidence that looser policy reduces risk aversion and uncertainty. Gómez-Cram and Grotteria (2022) explore the price discovery process for several asset classes on FOMC announcement days. Bauer et al. (2022) develop a policy uncertainty measure based on financial derivatives, and show that FOMC (uncertainty cycle) announcements reduce uncertainty. Finally, Hattori et al. (2016) study the impact of Unconventional Monetary Policy (UMP) on stock market and bond market tail risk. UMP increases (decreases) the realized volatility of stocks (bonds), but lowers the tail risk in both markets. Forward guidance (and hence communication) appears to have stronger "dampening effects", compared to other UMP events.

We extend this line of research in two ways. First, these aforementioned studies often overlook extreme market responses when assessing the effects of news. For example, the main result of Hattori et al. (2016) that UMP decreases the tail risk in stock and bond markets does not appear to hold when we move outside the cycles of FOMC press releases. Unlike Hattori et al. (2016), we focus on the intraday market responses to speeches, which can occur at any time, rather than only the times of FOMC announcements, and measure the *realized* tail risk instead of the *implied* tail risk from derivatives. In contrast with Hattori et al. (2016), we find that speeches *increase* realized tail risk. This type of CBC does not appear to reduce uncertainty and calm financial markets.

Second, prior research on monetary policy news has commonly employed jump-diffusion models with Poisson jumps to capture responses to news. The approach of Bauer et al. (2022) relies on such a representation for "FOMC jumps". Despite its simplicity, these jump models are not compatible with the stylized facts of jump occurrences, as news-induced tail responses are persistent in the presence of heterogeneous investors interpreting the content of speeches. Consequently, these studies underestimate the realized tail risk. Departing from this conventional approach, we consider a more flexible model that allows for *time-varying* tails. This allows us to separate extreme volatility responses from the tail responses and, more importantly, to identify the speeches that create *tail cascades*. Unlike the previous studies treating jumps as one-shot events, we accommodate the stochastic intensity of jumps that potentially occurs from heterogeneous interpretation of news by market participants. Our high-frequency event study approach is hence more flexible methodologically and better captures the dynamics of intradaily volatility and tail risk.

## Regime Dependence of Monetary Policy Effects

Both theory and data suggest that monetary policy is regime dependent. Mandler (2012) uses a threshold vector autoregression (VAR) framework to analyse the effectiveness of classical monetary policy shocks, depending on the respective inflationary regime in the US economy between 1965-2007. He finds that monetary policy shocks have markedly different effects in low and high inflation regimes. Such inflation regime differences can be theoretically motivated. Sizeable deviations from inflation target levels might affect a central bank's credibility and its ability to credibly signal. Similarly, substantial off-target inflation levels might affect private sector inflation expectations, altering the Philips curve and inflation dynamics (Mandler, 2012).

Tenreyro and Thwaites (2016) examine GDP regime dependence of monetary policy shock effects, derived from the unexpected component of interest rate changes. The empirical results of Tenreyro and Thwaites suggest that medium- to long-run monetary policy shock effects on the real economy strongly depend on the state of the business cycle. GDP growth is the most consistent factor determining monetary policy effectiveness, and shocks seem to have a more pronounced effect during economic upswings than during downswings.[6] They also

[6] Tenreyro and Thwaites (2016) further emphasize the historical evidence that fiscal policy measures have been more important in times of recession, while fiscal and monetary policy have historically reinforced one

find that contractionary shocks have greater impact than expansionary ones, with both being equally represented during recessions and booms. Desired effects of policy rate changes might be subdued during recessions and central bankers might rely more strongly on unconventional monetary policy near the effective lower bound (ELB). To the best of our knowledge, we are the first to investigate regime dependence — with regards to both inflation and GDP growth — of the effectiveness of unconventional monetary policy and central bank communication.

### Text Analysis for Monetary Policy

Lastly, we are part of a burgeoning literature that uses natural language processing to analyse monetary policy. Various text analysis methods have been tested in this field. For example, researchers have used topic models (Hansen et al., 2019), combined dictionary methods with classic machine learning models such as XGBoost (Petropoulos and Siakoulis, 2021), and have deployed deep neural network models such as transformers (Cai et al., 2021). In our work, instead of choosing a specific NLP algorithm a priori, we decide to take a more model-agnostic, data-driven approach to reduce modeler bias. That is, we train a variety of NLP models and choose the algorithm that works best in our validation set.

Similarly, researchers have employed various frameworks and datasets to identify monetary policy news. In particular, researchers have often studied the market effects of central bank policy announcements. For instance, Lucca and Trebbi (2009) and Hansen and McMahon (2016) both leverage approaches from computational linguistics within a VAR framework to asses the effect of the content in FOMC statements on macroeconomic variables. Lucca and Trebbi (2009) find CBC to be a more important factor than contemporaneous policy rate decisions. Hansen and McMahon (2016) conclude that shocks to forward guidance have a stronger effects on markets than communication of current economic conditions. Handlan (2020) uses a deep neural network architecture to identify text-based shocks in FOMC announcements, assessing their impact on Fed funds futures. She finds that shocks derived from forward guidance wording of FOMC statements account for four times more variation in Fed funds future prices than direct announcements of changes in the target federal funds rate. Gómez-Cram and Grotteria (2022) apply a video analysis on words mentioned during central bank press conference videos. Nesbit

another during booms.

(2020) proposes a word count based instrumental variable framework to identify monetary policy shocks in FOMC transcripts. Aruoba and Drechsel (2022) use NLP techniques to analyse FOMC meetings in order to measure the information set of the FOMC at the time of policy decisions. They then use these measures to generate estimates of FOMC monetary policy shocks.

Although each of these studies use different methods, they all utilise text to help us to identify effects of monetary policy. However, official central bank announcements, such as FOMC announcements, occur only infrequently (every 6-8 weeks). We therefore shift our focus on central bankers' speeches which happen in much higher frequency. Researchers have paid only limited attention to speeches, partly because their content is difficult to quantify. At the same time, central bank deliberation and communication is continuous (Neuhierl and Weber, 2019). Thus, it is important to frequently measure CBC effects.

A few notable papers move in this direction. Neuhierl and Weber (2019) find that the tone of US Fed chair and vice-chair speeches, measured via word count methods, can explain stock market price dynamics. Using a mixture of machine learning and dictionary methods, Petropoulos and Siakoulis (2021) derive sentiment indices from central bank speeches and find that the sentiment predicts financial turmoil. We use a two-step macroeconomic news identification framework, in which we first learn a mapping from central bank language to central bank forecasts with Greenbook data, and then infer how FOMC member speeches imply revisions to GDP, inflation, and unemployment forecasts — an approach which is motivated by Ahrens and McMahon (2021).

To identify the news content of a speech, we must control for market expectations. Ellen et al. (2022), for example, construct a monetary news series from the difference in narrative between central bank statements and news media coverage. The results of Ellen et al. (2022) highlight the pivotal role of news media as catalysts in the process of forming market expectations and confirm earlier findings in the literature that monetary policy shocks cause measurable macroeconomic responses. Similarly, Cai et al. (2021) analyse FOMC announcements using BERT (Devlin et al., 2018) and identify monetary policy and information shocks, controlling for market expectations by analysing relevant New York Times articles with NLP methods. Instead of inferring market expectations from noisy news media coverage, we take the latest forecast measures from the widely viewed Survey of Professional Forecasters (SPF) conducted by the

Federal Reserve Bank of Philadelphia. SPF forecasts directly measure expected GDP growth, inflation, and unemployment. We then define a macroeconomic news shock as the difference between a speech-implied forecast revision and the most recent SPF forecast for that variable available at the time of the speech.

## 6.4 Federal Reserve and Markets Data

The data used in our paper consists of several types: FOMC member speeches, Greenbook text, Greenbook forecasts, SPF forecasts, and intraday volatility and tail risk measures of US stock and bond markets. We use Greenbook forecasts and the respective Greenbook text sections that describe them to map central bank language to central bank forecasts. We then apply our learned mapping to FOMC member speeches and assess how speech-implied forecast revisions affect volatility and tail risk in financial markets.

### 6.4.1 Federal Reserve Speech and Forecast Data

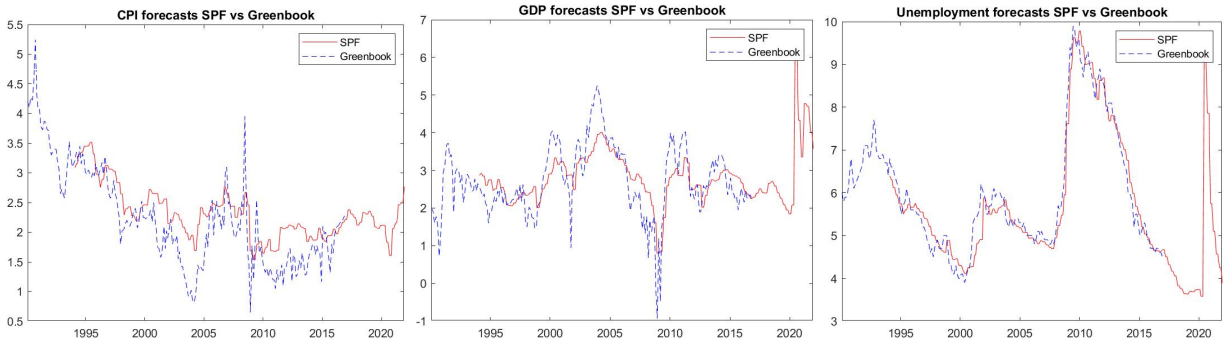
The central bank data is split into a training and a test set. We describe these datasets below.

**Training set:** In the training phase, we learn the mapping of the Fed's Greenbook texts associated with the descriptions of GDP growth, CPI, and unemployment outlooks to the change in the Greenbook forecasts of those variables from the previous forecast period. That is, we target the difference in a current period's one-quarter-ahead Greenbook forecast to the previous quarter's forecast, such that for any of our macroeconomic key figures of interest, $y$, we define $\Delta y_m = y_m - y_{m-1}$, where $m$ indicates the date of the Greenbook forecast. We also tested a one-year-ahead horizon, although this was less informative as one-year forecasts tend to revert to long-run values. The training sample spans 145 Greenbook documents, from January 1, 1995 to December 31, 2013. We only consider the 8,155 Greenbook sections that directly relate to GDP growth, CPI, and unemployment (see Appendix C.1 for a detailed list of section allocations). The average Greenbook section in our dataset has about $3,000$ words; the longest section consists of $31,000$ words and the shortest section contains around 140 words. At any date, we concatenate all Greenbook sections that relate to the same forecasting variable.

**Test set:** Training the NLP models consists of estimating complex mappings from Greenbook text on each date, for each variable, to the associated revisions to the one-quarter-ahead

Greenbook forecasts on each date, for each variable. Once the models are trained, we apply the learned mappings to a test set consisting of FOMC members' speeches made from January 1, 2014 to December 31, 2021. The applied mappings imply one-quarter-ahead forecast revisions for GDP growth, CPI, and unemployment. We assume that central bankers' speeches convey news from the Fed's information set that can alter the economic outlook of private agents. The Fed's information set could contain private or superior information about economic conditions, superior or alternative analysis (as in Byrne et al., 2023), or new information about the Fed's own preferences for monetary policy.

Figure 6.1: Comparison of Greenbook and SPF forecasts




*The figure displays the Greenbook and SPF forecasts over time for CPI (left panel), GDP (middle panel) and unemployment (right panel). The two forecasts match quite closely for the majority of the inspected time-series.*

The target variables in the test set are the one-quarter-ahead respective changes in GDP growth, CPI, and unemployment in the SPF forecasts. The SPF is a publicly available and widely referenced source for economic forecasts. We use the SPF as our proxy for market expectations, rather than the next Greenbook forecasts, because Greenbook forecasts are released to the public with a 5-year delay. We expect that central bank speeches should have similar predictive power for Greenbook and SPF forecast revisions. Figure 6.1 corroborates the assumption that the SPF forecasts match the Greenbook forecasts quite well during 1993 to 2016. We assume that this pattern also holds post 2016, for which there was no public Greenbook data available when the data for this paper was collected. We release our dataset of central bank speeches, time-stamped on the minute of release, on our Github repository[7].

[7] github.com/MaximilianAhrens/data/tree/main/central_bank_speeches

### 6.4.2 High-Frequency Market Data

We use high-frequency transaction prices for 22 Dow Jones Industrial Average (DJIA) stocks, together with 2-year, 5-year, and 10-year U.S. Treasury note and bond futures traded on the Chicago Board of Trade (CBOT). Appendix C.2 lists the individual stocks and bonds. Wharton Research Data Services (WRDS) and Tick Data LLC provide data for individual stocks and bond futures, respectively. As is standard in the literature, we exclude U.S. holidays, Christmas periods, and weekends from our sample. We only consider trading hours from 9:30 EST−16:00 EST and 7:30 CT−14:00 CT, for stock and bond markets, respectively. To reduce the potential impact of market microstructure noise, we filter out *bouncebacks* and irregular quotes that typically occur in ultra high-frequency data. Using our adjusted data, we create equally-spaced 15-second observations, which is an appropriate frequency to implement our response measures. Our sample runs from January 1, 2014 through December 31, 2021.

## 6.5 Methodological Framework

Our methodological framework can be broken down into two parts. Section 6.5.1 explains our multimodal NLP framework used to estimate the mapping from central bank language to forecasts. We test and compare our estimation framework with a variety of machine learning algorithms. Section 6.5.2 then describes the measurements of the asset price dynamics and their relationship with the speech signals.

### 6.5.1 Multimodal NLP Framework

We seek to estimate how new information revealed in central bank speeches influences financial markets. To do so, we map central bank language to macroeconomic forecasts, controlling for the macroeconomic conditions at the time.

The macroeconomic conditionality is important because the effect of a given forecast revision on financial markets depends on initial economic conditions. This economic context requires the multimodal modelling approach. For example, a speech that raised forecast inflation would be a positive signal of improving conditions if inflation was below its desired level. However, the same speech would convey a negative signal if inflation was substantially above target. We employ multimodal machine learning approaches that allow us to use both text and numeric

data when mapping central bank language to central bank forecasts and then predicting output, inflation, and unemployment outlook revisions.

#### 6.5.1.1 Learning Mapping from Central Bank Language to Forecasts

We learn the mapping from the Fed's Greenbook text to the respective Greenbook forecasts. The Greenbooks contain dedicated sections on the Fed's forecasts of GDP growth, CPI, and unemployment, including the rationales for the forecasts. These sections allow us to map the Greenbook text - ergo central bank language - to central bank forecasts.

In the training phase, we estimate a separate mapping for each of the three variables, i.e., the one-quarter-ahead forecast change in CPI, GDP growth, or unemployment. We measure the change from the previous $(m-1)$ Greenbook to the current $(m)$ in the one-quarter-ahead forecasts $(q_1)$. CPI is denoted by $\pi$, GDP growth by $g$, and unemployment by $u$. Hence, our three target variables are: $\Delta\pi_{q_1,m}$, $\Delta g_{q_1,m}$, and $\Delta u_{q_1,m}$. For ease of notation in the following equations of our modelling framework, let $y$ serve as a placeholder variable for any of the CPI, GDP growth, and unemployment variables. Hence, we denote our placeholder target variable as $\Delta y_{q_1,m}$.

To capture the economic context, we control for both change and level of the CPI, GDP, and unemployment of the previous Greenbook report, denoted as $X_{m-1}$. We fit a function, $f$, to learn how the respective Greenbook text maps into forecasts, controlling for macroeconomic conditions. The equations for CPI, GDP growth, and unemployment have the same explanatory variables, except for the text input, which is specific to the respective Greenbook forecast section. That is, $\theta_\pi$ represents the text features for the CPI corpus, while $\theta_g$ represents GDP-related text, and $\theta_u$ unemployment-related text. We use $\theta_y$ as a placeholder for any of the three text inputs. With this notation, $\theta_{y,k}$ represents the $k^{th}$ text feature for the respective target variable $y$. Let us define $f$ as the function that takes text and numeric data as inputs and maps them to the target output $y$, given parameters $\Omega$, which are to be learned. We can now write out our regression equation as

$$\Delta y_{q_1,m} = f\left(X_{m-1}, \theta_{y_m}; \Omega\right). \tag{6.1}$$

If we assume linearity in function $f$, the regression equation can be written as follows:

$$\begin{aligned}\Delta y_{q_1,m} = {} & \omega_\pi \pi_{q_1,m-1} + \omega_g g_{q_1,m-1} + \omega_u u_{q_1,m-1} \\ & + \omega_{\Delta u} \Delta u_{q_1,m-1} + \omega_{\Delta \pi} \Delta \pi_{q_1,m-1} + \omega_{\Delta g} \Delta g_{q_1,m-1} \\ & + \sum_{k=1}^{K} \omega_k \theta_{y,k,m} + \epsilon_m. \end{aligned} \tag{6.2}$$

Here, the $\omega$s represent the regression parameters and $\epsilon$ is the measurement error. We use the first 80% of the Greenbook dataset for training and the remaining 20% for validation. The data is furthermore de-meaned and standardized based on training set values. We did not randomly split the training and validation set to acknowledge the time-series characteristics (and therefore the potential for information leakage) in the data. We then train the machine learning models to map central bank texts and control variables to the respective target variables. We treat this as a regression problem and use a least squares error loss function, commonly used in economics and monetary policy econometrics.

#### 6.5.1.2 Identifying Information Signals in Central Bank Speeches

In the test phase, we apply the trained models for each of the macroeconomic variables (CPI, GDP growth, unemployment) to the central bank speeches to infer macroeconomic forecast revisions. The text data is now the central bank speech content. The numeric data points on current economic conditions are the most recent SPF forecast levels and changes on GDP growth, CPI, and unemployment.[8] This procedure maps each central bank speech into an implied revision of the forecasts for CPI, GDP growth, and unemployment.

#### 6.5.1.3 Calculating News Signals

Markets should only react to relevant news that have not yet been incorporated into asset prices. If a central bank speech does not change the expected macroeconomic path, then the speech has no news component. We proxy market expectations with the latest public SPF forecast for each target variable. We then calculate the difference between the most recent SPF forecast change ($\Delta y_{SPF,s}$) available at the time of each speech and the implied forecast change in each

[8] As previously shown in Figure 6.1, the SPF forecasts track the Greenbook forecasts quite closely.

speech ($\Delta\hat{y}_{\text{speech},s}$). This difference is our forecast revision news, $\nu$, for target variable, $y$, and speech event, $s$, such that

$$\nu_{y,s} = \Delta y_{\text{SPF},s} - \Delta\hat{y}_{\text{speech},s}. \tag{6.3}$$

For GDP, a positive difference, $\nu_{y,s}$, is bad news, because a positive value means that the central bank speech implies lower GDP growth than does the most recent SPF forecast. The opposite is true for unemployment. Here, a positive difference is good news, as the speech implies that the central bank expects unemployment rates to fall faster (or rise less quickly) than previously anticipated.

For CPI, the categorisation into good and bad news depends on the relation of the current inflation level to the target. The Fed aims for an inflation rate of around 2%, as do most central banks of advanced economies.[9] Therefore, a positive $\nu_{\pi,s}$ — i.e., an implied downward forecast revision — is good news when the forecast of inflation is above target. This means inflation will revert faster back to target than anticipated (or won't rise as fast as anticipated). Conversely, when forecast of inflation rate is below target, a negative $\nu_{\pi,s}$ is good news. A later analysis will assess how financial market volatility and tail risk react to these implied forecast-revisions.

#### 6.5.1.4 Machine Learning Methods

We do not know, a priori, which statistical learning model would best approximates the function, $f$, in equation (6.1). We have relatively few data points compared to many machine learning projects (e.g. hundreds or thousands rather than millions or billions of data points). Each data point itself is rich in information, however, consisting of a high dimensional feature set. That is, each set of text can be several thousand words long, which presents a problem for many modern language models such as transformer family models (e.g. BERT-based models), which can usually only handle up to around 100-1,000 tokens per data point (Das et al.). Some extensions based on sparse transformers have been proposed such as Child et al. (2019) and Zaheer et al. (2020), which can handle sequences of a couple of thousand tokens. However, document lengths of $20,000+$ words would still pose a challenge. Lacking reason to favour a

[9] The FOMC targets a 2% rate of change for the personal consumption expenditure price index (PCE), not the CPI. The two inflation rates are very highly correlated, however, which makes it reasonable to use information about implied CPI forecasts to proxy for PCE forecasts.

specific class of models, we deploy a range of models, to search broadly for the best model and reduce the a priori modeler bias of favouring one model over alternatives.

We therefore deploy an extensive array of multimodal machine learning algorithms to approximate function $f$ and to learn parameters $\Omega$. We use the multimodal machine learning benchmark suite, AutoGluon (AutoGL) (Erickson et al., 2020), and we add to it the class of multimodal supervised topic models (Card et al., 2018; Ahrens et al., 2021). The research task here is prediction and not causal inference. We therefore opt for rSCHOLAR instead of BTR. As shown in chapter 5, both models demonstrate comparable predictive performance. However, rSCHOLAR is more computationally efficient as it is solved via variational inference instead of Gibbs sampling.

### AutoGluon

AutoGL is an automated machine learning (AutoML) framework that has been developed to fuse multimodal features such as text, images, and numeric data. We chose this AutoML framework because it outperformed competing frameworks in multimodal benchmark tasks (see Erickson et al., 2020).

**Base models**: AutoGL fits machine learning *base models* and then combines them through ensembling and stacking to boost performance. AutoGL allows us to apply hyperparameter optimization over all models. The *base models* in AutoGL span the following broad machine learning algorithm classes:

1. **K-nearest neighbours** (Dudani, 1976): AutoGL uses two variations of k-nearest neighbours (KNN) that differ in their weighting approaches. One allocates uniform weights to all points while the other weights points according to the inverse of their respective distances.

2. **Random forests** (Breiman, 2001): AutoGL again deploys two variations of this algorithm class. One option uses the information gain of nodes for the assessment of the split quality. The other option uses Gini impurity instead.

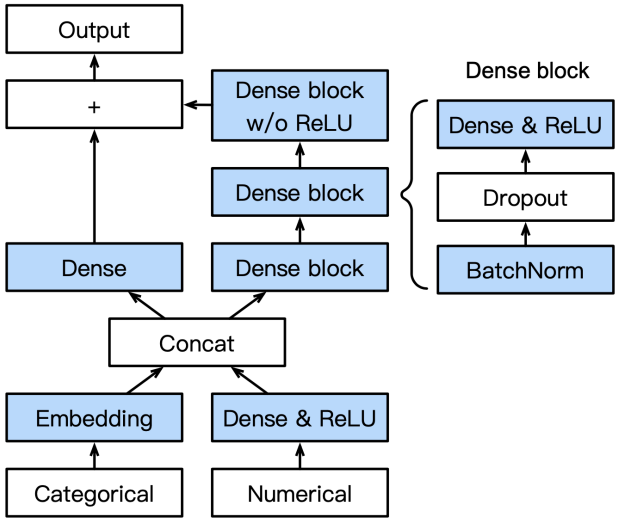
3. **Extremely randomized trees** (Geurts et al., 2006): For the random tree class, AutoGL deploys both an implementation resorting to information gain and another option that

uses Gini impurity for the assessment of split quality.

4. **Boosted decision trees**: AutoGL runs (where applicable to the task) Extreme Gradient Boosting (Chen and Guestrin, 2016), Light Gradient Boosting (Ke et al., 2017), Categorical Boosting (Prokhorenkova et al., 2018).

5. **Neural networks**: Figure 6.2 schematically outlines AutoGL's neural network architecture, which Erickson et al. (2020) details. The architecture has been specifically designed for the multimodal use of categorical (text, images) and numeric data. It uses variable-specific embeddings for each of the categorical features. These are then concatenated with the numeric features into one overall input vector. This vector is in turn fed through a 3-layer feed-forward network as well as through a linear skip-connection (for details see Erickson et al., 2020). Model ensembling and stacking can be applied and are optimally chosen in the validation process.

Figure 6.2: AutoGL schematic neural network architecture




*The figure displays the AutoGluon schematic neural network architecture, based on the design by Erickson et al. (2020), p. 3. Layers with learnable parameters coloured in blue.*

**Text representation options**: We must also choose how to represent the text in machine-readable format. We define the following approaches:

1. **AutoTab**: Only tabular numeric features are used. Text is excluded. AutoTab is our tabular numeric data baseline next to an OLS regression that only uses tabular numeric

data.[10]

2. **AutoTab + tfidf**: Use tf-idf weighted word counts of the text as features. Standard text cleaning procedures of removing stopwords and punctuation have been applied.

3. **AutoTab + topics**: Use topic shares from supervised topic models as features (using rSCHOLAR without tabular numeric data for the topic estimation).

4. **AutoMM transformer**: Use the AutoGL's multimodal modelling infrastructure that is based on a large language model (we use Roberta-base (Liu et al., 2019)) for multimodal fine-tuning. Tabular numeric data can be fused into this process as well.[11]

5. **AutoTab + embed**: Use AutoMM transformer as well as AutoTab models that featurize text data as n-grams and ensemble over this zoo of models.[12]

### 6.5.2 Asset Price Dynamics

#### 6.5.2.1 Underlying Continuous-Time Model

We model the intraday behaviour of asset prices with the following continuous-time model: The log-price $X$ of each asset (stock or bond) follows an Itô semimartingale defined on a filtered space $(\Omega, \mathcal{F}_t, (\mathcal{F}_t)_{t\in[0,T]}, \mathbb{P})$ over an interval $[0, T]$. The Grigelionis decomposition (see e.g., Erdemlioglu and Yang, 2022; Boswijk et al., 2018; Dungey et al., 2018) implies that $X_t$ has the following specification:

$$X_t = X_0 + \int_0^t b_s ds + \int_0^t \sigma_s dW_s + \delta * (\mu_t - \psi_t) + (\delta - h(\delta)) * \mu_t, \tag{6.4}$$

where $b_s$ is the drift term, $\sigma_s$ is the stochastic volatility component, $W$ is a standard Brownian motion, $\delta$ is a predictable function, $h$ is a truncation function (e.g., $h(x) = x1_{\{\|X\|\leq 1\}}$), $\mu$ is the jump measure of $X$, and $\psi$ is its jump compensator, which adopts the decomposition

$$\psi_t(dt, dx) = [f_t(x)\lambda_t dx]dt$$

[10] AutoGL's *TabularPredictor* approach.
[11] AutoGL's *MultimodalPredictor* approach.
[12] AutoGL's *TabularPredictor* approach with the *hyperparameter* option being set to *multimodal*.

where the function, $f_t(x)$, controls the jump size distribution and $\lambda_t$ denotes the jump intensity as in Erdemlioglu and Yang (2022) and Boswijk et al. (2018). We focus on the *tail* component of this jump compensator or $\lambda_t$, which captures the jump intensity dynamics.[13] We can specify $\lambda_t$ as

$$\lambda_t = \lambda_0 + \int_0^t b'_s ds + \int_0^t \sigma'_s dW_s + \int_0^t \sigma''_s dB_s + \delta' * \mu_t + \delta'' * \mu_t^{\perp}, \tag{6.5}$$

where $B$ is a standard Brownian motion independent of $W$, $\mu_t^{\perp}$ is orthogonal to $\mu_t$, and $\delta'$, $\delta''$ are predictable. This model, given by equations (6.4) and (6.5), satisfies no-arbitrage conditions and leaves the volatility and jump components unrestricted. We now present our volatility and tail risk measures from this model.

#### 6.5.2.2 High-Frequency Measurement of Volatility and Tail Risk

Given the price dynamics in equations (6.4) and (6.5), let us define the $i$th intradaily return on a trading day as $r_{i,t} = X_{i,t}$ - $X_{i-1,t}$. We can write the daily realized volatility ($RV$) as the square root of realized variance, which is the sum of the squared intraday returns $(1, \ldots, M)$. That is,

$$RV = \sqrt{\sum_{i=1}^{M} r_i^2}. \tag{6.6}$$

It is well-known that realized variance converges to quadratic variation (see e.g., Andersen et al., 2003, 2001 and Barndorff-Nielsen and Shephard, 2002 for in-depth discussion).

Turning to the estimation of $\lambda_{i,t}$ in equation (6.5), we define the post-signal realized intensity ($RI$) measure as

$$RI = \frac{\Delta_n^{\varpi \widehat{\beta}_i}}{k_n \Delta} \sum_{j=1}^{k_n} g\left(\frac{|r_i|}{\alpha \Delta^{\varpi}}\right) \frac{\alpha^{\widehat{\beta}}}{C_{\widehat{\beta}_i}(k_n)}, \tag{6.7}$$

where $\Delta$ is incremental change between observations, $\alpha\Delta^{\varpi}$ is threshold to retain only large jumps, $g(\cdot)$ admits a specific functional form, $k_n$ is a constant which admits $(1/K \leq k_n \Delta^{\rho} \leq K)$ for $(0 < \rho < 1)$ and $(0 < K < \infty)$, and $\beta_i$ is the estimator of jump activity index that controls the vibrancy of sharp fluctuations. In 6.7, $g(\cdot)$ as an auxiliary function that separates jump-type movements from the diffusive volatility, based on an $\alpha$ deviation (e.g., $\alpha = 2, 3, 6$) from the continuous component of the model.[14] We use $RI$ as a proxy for time-varying (high-frequency)

[13] See Andersen et al. (2020), who exploit jump intensity process to measure tail risk and assess its equity premium implications.

[14] See e.g., Erdemlioglu and Yang (2022), Boswijk et al. (2018) and Dungey et al. (2018) for implementation

*tail risk* ($TR$), which is considerably accurate at high frequency, similar to the measures adapted in **?**.[15]

In summary, we quantify two types of responses to CBC. First, communication likely creates sudden surges in market volatility. We assess these surges with realized volatility. Second, CBC can cause asset price jumps and persistently elevated jump intensity. Our approach allows us to first detect the speech-implied jumps, and then assess the 'intensity' of the jump responses. As Bollerslev et al. (2018) document, heterogeneous investors often release private information as they trade in the wake of such jumps, creating large price moves, which amplify high-frequency $TR$.

#### 6.5.2.3 Identifying Association Between News and Market Reactions

The final step in our methodological framework is to measure how realized volatility and tail risk in both equity and bond markets react to central bankers' speeches. To this end, we regress the market reactions on the forecast revision implied by the corresponding speech. As the forecast revision itself is a linear combination of the central bank signal and the latest public forecast, we already control for the partial correlation between the SPF forecasts and the market reactions.[16] The same holds true for all control variables used in the creation of the speech signals. We don't add additional low-frequency macroeconomic control variables because market prices should already incorporate such publicly available information.

## 6.6 Results: Language Mapping and SPF Prediction

*Refer to updated and improved section in the published version in the Journal of Econometrics (December 2024)*

The first step of our method is to learn the mapping from central bank language to central bank forecasts. We train our model on the first 80% of the Greenbook sample, holding out the last 20% of observations for validation. In our validation set, we assess how well a model can map Greenbook language to Greenbook forecasts. For each machine-learning class, we select the best performing model from the validation set and then assess its performance on the test

details, particularly on the selection of the functional form for $C_{\widehat{\beta}_i}(k_n)$ in (6.7).

[15] Our tail risk indicator $RI$ is also quite similar to the estimator of Hill (1975). See also Aït-Sahalia and Jacod (2009) for a related discussion on the role of $\beta_i$ in (6.7) when capturing tails of return distributions.

[16] See Frisch-Waugh-Lovel theorem such as Frisch and Waugh (1933) and Lovell (1963).

set. The test sample is the post-2013 sample of speeches in which we assess how well the speech signals predict subsequent changes in SPF forecasts. Given the results in the Tables 6.1, 6.2, and 6.3, we have reason to believe that the identified signals in the central bank speeches carry relevant information to change market expectations and hence public macroeconomic forecasts. The tables report the $R^2$ associated with predictions of SPF forecast revisions.

For example, the second row of Table 6.1 indicates that the multimodal neural topic model (MM NTM non-linear) has an $R^2$ of 0.67 in predicting CPI forecast revisions in the Greenbook training set, 0.83 in the Greenbook validation set, and 0.735 in the test set (speeches). Appendix C.3 shows all tested machine learning approaches.

For each of the three macroeconomic target variables, the best multimodal NLP models markedly outperform models that only use numeric data. Specifically, the multimodal neural topic model (MM NTM) class performs best both in the validation and in the test set. For CPI, Table 6.1 shows that the MM NTM (non-linear) model has an $R^2$ of 0.735 in the test set, which is 15% better than MM NTM (linear) and 44% better than the $R^2$ of the next best method. Likewise, Table 6.2 shows that MM NTM (non-linear) has an $R^2$ of 0.797 in the test set, which is right behind MM NTM (linear)'s $R^2$ of 0.825. Finally, Table 6.3 shows that MM NTM (non-linear) performs best again for unemployment, with an $R^2$ of 0.208, which is markedly better than the second best $R^2$ of 0.131, achieved by AutoTab.

Interestingly, AutoGL's models underperform an OLS regression for CPI inflation and GDP growth. There might be several explanations for this underperformance. The datasets at hand contain relatively few data points — a common challenge in macroeconomics and macro-finance, especially for 'data hungry' machine learning methods. AutoGL's machine learning models might therefore struggle to converge or might easily overfit on the limited training data. Second, macroeconomic forecasts (or the revisions to them) might be well approximated by a linear model, since such models are a very common design choice in monetary economics, macroeconomics, and macroeconometrics. Hence, perhaps the relatively strong performance of an OLS regression compared to the AutoGL models.

Table 6.1: Central Bank Language to Forecast Mapping - CPI Q1

| **Metric: $R^2$** | **train (GB)** | **val (GB)** | **test (speeches)** |
|---|---|---|---|
| OLS | 0.288 | | 0.510 |
| MM NTM (linear) | 0.600 | 0.650 | 0.640 |
| **MM NTM (non-linear)** | 0.670 | **0.830** | **0.735** |
| AutoTab | 0.565 | 0.302 | 0.475 |
| AutoTab + tfidf | 0.953 | 0.305 | 0.299 |
| AutoTab + topics | 0.370 | 0.284 | 0.358 |
| AutoTab + embed | 0.573 | 0.139 | 0.132 |
| AutoMM transformer | -0.155 | -† | -0.292 |

*The table reports $R^2$ for training, validation, and test sets for each of the models. Best performing model in validation and test set in bold. †: Model only reports MSE for validation set.*

Table 6.2: Central Bank Language to Forecast Mapping - GDP Q1

| **Metric: $R^2$** | **train (GB)** | **val (GB)** | **test (speeches)** |
|---|---|---|---|
| OLS | 0.301 | | 0.785 |
| **MM NTM (linear)** | 0.372 | **0.426** | **0.825** |
| MM NTM (non-linear) | 0.483 | 0.371 | 0.797 |
| AutoTab | 0.497 | 0.304 | 0.380 |
| AutoTab + tfidf | 0.752 | 0.240 | 0.268 |
| AutoTab + topics | 0.730 | 0.253 | 0.285 |
| AutoTab + embed | 0.587 | 0.220 | 0.142 |
| AutoMM transformer | 0.013 | -† | -0.044 |

*The table reports $R^2$ for training, validation, and test sets for each of the models. Best performing model in validation and test set in bold. †: Model only reports MSE for validation set.*

Table 6.3: Central Bank Language to Forecast Mapping - Unemployment Q1

| **Metric: $R^2$** | **train (GB)** | **val (GB)** | **test (speeches)** |
|---|---|---|---|
| OLS | 0.231 | | -0.377 |
| MM NTM (linear) | 0.197 | 0.109 | 0.066 |
| **MM NTM (non-linear)** | 0.285 | **0.457** | **0.208** |
| AutoTab | 0.191 | 0.058 | 0.131 |
| AutoTab + tfidf | 0.577 | 0.113 | -0.045 |
| AutoTab + topics | 0.278 | 0.053 | -0.010 |
| AutoTab + embed | 0.415 | 0.145 | -0.044 |
| AutoMM transformer | -0.737 | -† | -1.177 |

*The table reports $R^2$ for training, validation, and test sets for each of the models. Best performing model in validation and test set in bold. †: Model only reports MSE for validation set.*

## 6.7 Results: Intraday Market Effects

*Refer to updated and improved section in the published version in the Journal of Econometrics (December 2024)*

We use the model that performed best in the validation set (Greenbook data) to estimate the speech-implied information on GDP, CPI, and unemployment forecast revisions in the test set (speech data). The news on forecast revisions, as outlined in section 6.5.1.3, are defined as the difference between the speech-implied forecast for CPI, GDP, and unemployment outlook and the respective most recent SPF forecast. We then fit an OLS regression where we use the speech-implied news as independent variables. Market volatility and tail risk are the respective dependent variables. We first show our estimation results across regimes in section 6.7.1. In section 6.7.2, we then segment our speech dataset into low, normal, and high GDP and CPI regimes, respectively. Section 6.7.3 shows the news effect analysis by CPI regime. Section 6.7.4 covers the same analysis by GDP regime.

### 6.7.1 News Effects Across Regimes

We use the estimated realized volatility ($RV$) and tail risk ($TR$) in the 30-minute window after a speech as as our dependent variables. We regress both $RV$ and $TR$ on all absolute speech-implied news across all regimes. That is, we expect larger forecast revision news (in absolute value) to raise volatility and tail risk. The data is de-meaned and standardized. For each speech $s$, denote its CPI news component as $\nu_{\pi,s}$, GDP news as $\nu_{g,s}$, and unemployment news as $\nu_{u,s}$. The regression equations for realized volatility and tail risk are then

$$RV_s = \beta_0|\nu_{\pi,s}| + \beta_1|\nu_{g,s}| + \beta_2|\nu_{u,s}| + \epsilon_{RV} \tag{6.8}$$

$$TR_s = \rho_0|\nu_{\pi,s}| + \rho_1|\nu_{g,s}| + \rho_2|\nu_{u,s}| + \epsilon_{TR}. \tag{6.9}$$

We estimate both equations for both equity and bond markets.

#### Equity Markets

The positive and statistically significant coefficients in the top panel of Table 6.4 reveal that larger absolute forecast revision news, i.e., larger absolute differences between the implied forecast and the most recent SPF forecast, are associated with higher realized equity volatility. All three types of forecast revisions are highly statistically significant at the 10% level. The bottom panel of Table 6.4 indicates that the magnitude of speech-implied forecast revisions to

CPI and unemployment has a statistically significant association with higher tail risk in equity markets. GDP news have no statistically significant effect.

Table 6.4: Association between absolute speech-implied forecast revision news and volatility (top panel) and tail risk (bottom panel) in equity markets across all regimes

| Target variable: $RV_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \|CPI news\| | 0.1675 | 0.022 | 7.585 | 0.000 | 0.124 | 0.211 |
| \|GDP news\| | 0.0780 | 0.043 | 1.800 | 0.072 | -0.007 | 0.163 |
| \|U news\| | 0.1967 | 0.024 | 8.078 | 0.000 | 0.149 | 0.244 |
| $R^2$: 0.722 | Adj. $R^2$: 0.718 | n. obs.: 191 | Heteroscedasticity robust standard errors | | | |
| Target variable: $TR_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
| \|CPI news\| | 2.2613 | 0.483 | 4.677 | 0.000 | 1.314 | 3.209 |
| \|GDP news\| | 1.1819 | 0.990 | 1.193 | 0.233 | -0.759 | 3.123 |
| \|U news\| | 2.4452 | 0.484 | 5.056 | 0.000 | 1.497 | 3.393 |
| $R^2$: 0.526 | Adj. $R^2$: 0.519 | n. obs.: 191 | Heteroscedasticity robust standard errors | | | |

*The table shows the association between speech-implied forecast revision news in absolute value about CPI, GDP, and unemployment and realized volatility (top panel) and tail risk (bottom panel). The estimation results are reported for the U.S. equity market.*

### Bond Markets

Tables 6.5, 6.6, and 6.7 show the results for the 2-, 5-, and 10-year bond futures markets. The bond market results are similar to those of the equity market. Larger absolute speech-implied forecast revision news are strongly associated with higher realized bond price volatility and tail risk across maturities.

Table 6.5: Association between absolute speech-implied forecast revision news and volatility (top panel) and tail risk (bottom panel) in bond markets (2-year maturity) across all regimes

| Target variable: $RV_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \|CPI news\| | 0.0149 | 0.003 | 5.643 | 0.000 | 0.010 | 0.020 |
| \|GDP news\| | 0.0110 | 0.005 | 2.121 | 0.034 | 0.001 | 0.021 |
| \|U news\| | 0.0166 | 0.003 | 5.412 | 0.000 | 0.011 | 0.023 |
| $R^2$: 0.672 | Adj. $R^2$: 0.667 | n. obs.: 175 | Heteroscedasticity robust standard errors | | | |
| Target variable: $TR_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
| \|CPI news\| | 3.7368 | 0.809 | 4.619 | 0.000 | 2.151 | 5.322 |
| \|GDP news\| | 5.4056 | 1.022 | 5.288 | 0.000 | 3.402 | 7.409 |
| \|U news\| | 3.3025 | 0.887 | 3.725 | 0.000 | 1.565 | 5.040 |
| $R^2$: 0.508 | Adj. $R^2$: 0.500 | n. obs.: 175 | Heteroscedasticity robust standard errors | | | |

*The table shows the association between speech-implied forecast revision news in absolute value about CPI, GDP, and unemployment and realized volatility (top panel) and tail risk (bottom panel). The estimation results are reported for 2-year maturity U.S. Treasury bond futures.*

Table 6.6: Association between absolute speech-implied forecast revision news and volatility (top panel) and tail risk (bottom panel) in bond markets (5-year maturity) across all regimes

| Target variable: $RV_{b,5y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \|CPI news\| | 0.0298 | 0.006 | 4.866 | 0.000 | 0.018 | 0.042 |
| \|GDP news\| | 0.0238 | 0.013 | 1.852 | 0.064 | -0.001 | 0.049 |
| \|U news\| | 0.0354 | 0.006 | 5.900 | 0.000 | 0.024 | 0.047 |
| $R^2$: 0.592 | Adj. $R^2$: 0.588 | n. obs.: 175 | Heteroscedasticity robust standard errors | | | |
| Target variable: $TR_{b,5y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
| \|CPI news\| | 2.3726 | 0.744 | 3.189 | 0.001 | 0.914 | 3.831 |
| \|GDP news\| | 3.6080 | 1.500 | 2.405 | 0.016 | 0.667 | 6.549 |
| \|U news\| | 1.4576 | 0.684 | 2.132 | 0.033 | 0.118 | 2.797 |
| $R^2$: 0.424 | Adj. $R^2$: 0.413 | n. obs.: 175 | Heteroscedasticity robust standard errors | | | |

*The table shows the association between speech-implied forecast revision news in absolute value about CPI, GDP, and unemployment and realized volatility (top panel) and tail risk (bottom panel). The estimation results are reported for 5-year maturity U.S. Treasury bond futures.*

Table 6.7: Association between absolute speech-implied forecast revision news and volatility (top panel) and tail risk (bottom panel) in bond markets (10-year maturity) across all regimes

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \|CPI news\| | 0.0574 | 0.010 | 5.687 | 0.000 | 0.038 | 0.077 |
| \|GDP news\| | 0.0443 | 0.021 | 2.132 | 0.033 | 0.004 | 0.085 |
| \|U news\| | 0.0614 | 0.010 | 6.000 | 0.000 | 0.041 | 0.082 |
| $R^2$: 0.650 | Adj. $R^2$: 0.644 | n. obs.: 175 | Heteroscedasticity robust standard errors | | | |
| Target variable: $TR_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
| \|CPI news\| | 1.8245 | 0.644 | 2.833 | 0.005 | 0.562 | 3.087 |
| \|GDP news\| | 3.0200 | 1.413 | 2.137 | 0.033 | 0.250 | 5.790 |
| \|U news\| | 1.3404 | 0.555 | 2.414 | 0.016 | 0.252 | 2.429 |
| $R^2$: 0.434 | Adj. $R^2$: 0.424 | n. obs.: 175 | Heteroscedasticity robust standard errors | | | |

*The table shows the association between speech-implied forecast revision news in absolute value about CPI, GDP, and unemployment and realized volatility (top panel) and tail risk (bottom panel). The estimation results are reported for 10-year maturity U.S. Treasury bond futures.*

### 6.7.2 Economic Regime Definitions

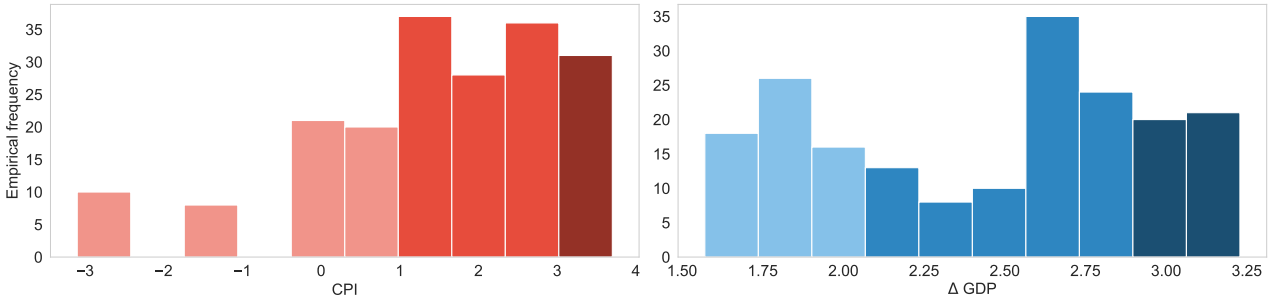
We also assess whether the effects of speech-implied forecast revisions depend on the GDP and inflation regimes. We do not separately analyse unemployment regimes. We divide our GDP and CPI datasets into a *high*, *normal*, and *low* regime (see Table 6.8). The categorisation is based on the Federal Reserve's inflation target and the historic distributions of the respective variables as depicted in Figure 6.3. Figure 6.4 shows the two time-series of the regime indicators.

Table 6.8: Categories of economic regimes

| | **CPI** | **ΔGDP** |
|---|---|---|
| **High** | $\pi > 3\%$ | $g > 3\%$ |
| **Normal** | $1\% < \pi < 3\%$ | $2\% < g < 3\%$ |
| **Low** | $\pi < 1\%$ | $g < 2\%$ |

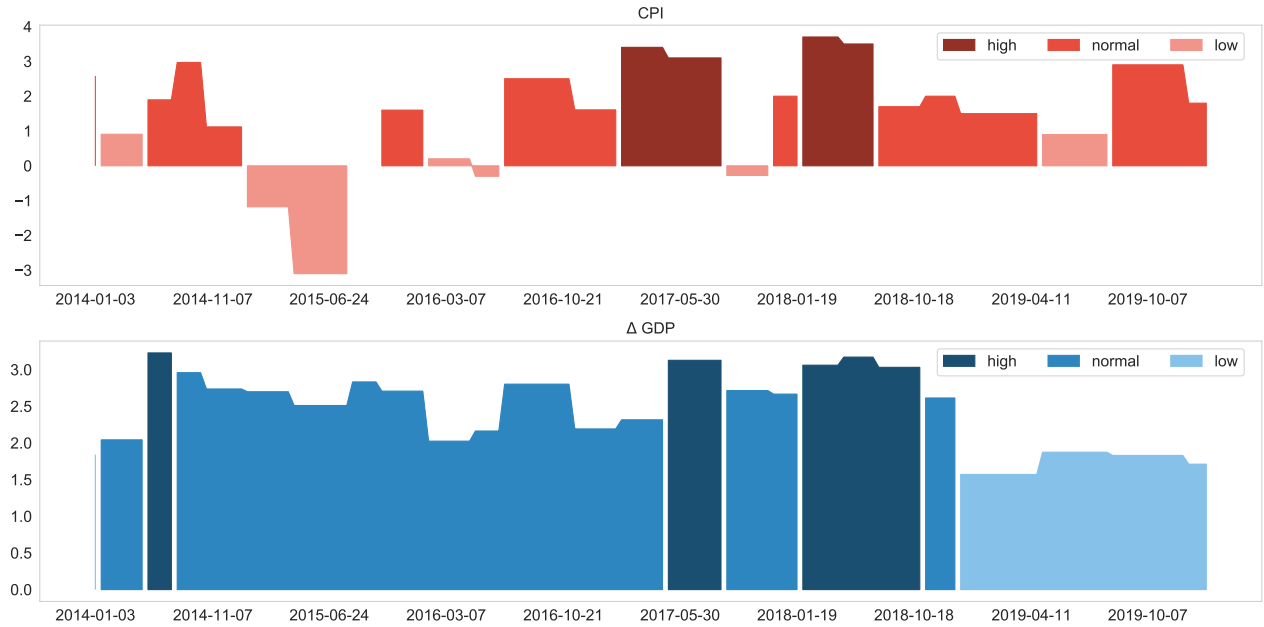
*The table presents the classification of different economic regimes (high, normal, low) for GDP and CPI.*

Figure 6.3: Empirical distribution of CPI and GDP growth target variables




*The figure shows the empirical distribution of CPI and GDP regimes. CPI: low regime (light red), normal regime (mid red), high regime (dark red). GDP: low regime (light blue), normal regime (mid blue), high regime (dark blue).*

Figure 6.4: Time-series of CPI and GDP growth regimes




*The figure displays the evolution of different economic regimes over time. CPI (upper panel): low regime (light red), normal regime (mid red), high regime (dark red). GDP (lower panel): low regime (light blue), normal regime (mid blue), high regime (dark blue).*

Conditional on the regime classification, we categorise the speech-implied news into *good* and *bad* news for the market. The division in the GDP-regime is straightforward. In any GDP regime, speeches that imply higher (lower) GDP-growth than the most recent SPF forecast are good (bad) GDP news. Similarly, lower (higher) unemployment forecast revisions are good (bad) new. The story for the CPI regime is more complex: If a speech implies that inflation will move closer to the 2% target than the most recent SPF forecast, it is considered good news. If a speech implies that inflation will move further from the target, it is bad news. So, a speech that implies an increase in inflation would be good news if inflation is below target but bad

news if inflation is above target. Table 6.9 outlines the news classifications.

Table 6.9: Central bank GDP news classification

| | **Good news** | **Bad news** |
|---|---|---|
| **High GDP** | $g_{\text{cb}} > g_{\text{spf}}$ | $g_{\text{cb}} < g_{\text{spf}}$ |
| **Normal GDP** | $g_{\text{cb}} > g_{\text{spf}}$ | $g_{\text{cb}} < g_{\text{spf}}$ |
| **Low GDP** | $g_{\text{cb}} > g_{\text{spf}}$ | $g_{\text{cb}} < g_{\text{spf}}$ |

*The table presents the classification of* good *versus* bad *GDP news for different levels of GDP.*

Table 6.10: Central bank CPI news classification

| | **Good news** | **Bad news** |
|---|---|---|
| **High CPI** | $\pi_{\text{cb}} < \pi_{\text{spf}}$ | $\pi_{\text{cb}} > \pi_{\text{spf}}$ |
| **Normal CPI (slightly above target)** | $\pi_{\text{cb}} < \pi_{\text{spf}} \vert \pi_{\text{spf}} > 2\%$ | $\pi_{\text{cb}} > \pi_{\text{spf}} \vert \pi_{\text{spf}} > 2\%$ |
| **Normal CPI (slightly below target)** | $\pi_{\text{cb}} > \pi_{\text{spf}} \vert \pi_{\text{spf}} < 2\%$ | $\pi_{\text{cb}} < \pi_{\text{spf}} \vert \pi_{\text{spf}} < 2\%$ |
| **Low CPI** | $\pi_{\text{cb}} > \pi_{\text{spf}}$ | $\pi_{\text{cb}} < \pi_{\text{spf}}$ |

*The table presents the classification of* good *versus* bad *CPI news for different levels of CPI.*

### 6.7.3 News Effects by CPI Regime

We now analyse the effects of speech-implied forecast revision news by CPI regime. We separate god news from bad news to assess whether asymmetric speech-implied news effects exist. The regression equations for realized volatility ($RV$) and tail risk ($TR$) in the 30 minutes after each speech are as follows:

$$RV_s = \beta_0|\nu_{\pi,s,good}| + \beta_1|\nu_{\pi,s,bad}| + \beta_2|\nu_{g,s,good}| + \beta_3|\nu_{g,s,bad}| + \beta_4|\nu_{u,s,good}| + \beta_5|\nu_{u,s,bad}| + \epsilon_{RV} \tag{6.10}$$

$$TR_s = \rho_0|\nu_{\pi,s,good}| + \rho_1|\nu_{\pi,s,bad}| + \rho_2|\nu_{g,s,good}| + \rho_3|\nu_{g,s,bad}| + \rho_4|\nu_{u,s,good}| + \rho_5|\nu_{u,s,bad}| + \epsilon_{TR}. \tag{6.11}$$

The variables have the same meaning as before. That is, for each speech $s$, denote its CPI news component as $\nu_{\pi,s}$, GDP news as $\nu_{g,s}$, and unemployment news as $\nu_{u,s}$. However, for each macroeconomic news component, we now have a *good news* variable and a *bad news* variable (both in absolute values), denoted by *good* and *bad* subscripts. We estimate the volatility

regression for both the equity and the bond markets for each CPI regime: low, normal, and high. The tail risk equation is estimated by CPI regime for equity markets only, due to scope limitations of this paper.

### Equity Markets

Table 6.11 reports the effects of speech-implied forecast revisions on realized volatility and tail risk in equity markets, broken down by CPI regime. Appendix C.4 details these results for each CPI regime and target variable.

Table 6.11: Association between speech-implied forecast revisions and volatility in equity markets across CPI regimes

| | **High CPI regime** | | **Low CPI regime** | | **Normal CPI regime** | |
|---|---|---|---|---|---|---|
| | RV | TR | RV | TR | RV | TR |
| \|News CPI good\| | $+^{***}$ | - | $+^{***}$ | - | - | $+^{*}$ |
| \|News CPI bad\| | $+^{***}$ | - | $+^{**}$ | - | - | - |
| \|News GDP good\| | - | - | $+^{**}$ | $+^{***}$ | - | - |
| \|News GDP bad\| | - | - | - | $+^{*}$ | - | - |
| \|News U good\| | $+^{***}$ | $+^{***}$ | - | - | - | - |
| \|News U bad\| | $+^{**}$ | - | $+^{***}$ | - | $+^{***}$ | $+^{***}$ |
| n. obs. | 36 | | 59 | | 70 | |

*$+$ = positive association. $^{*}= p \leq 0.1$, $^{**}= p \leq 0.05$, $^{***}= p \leq 0.01$. $-$ = no statistically significant results.*

**High CPI regime**: When CPI is high, speech-implied forecast revisions to CPI and unemployment forecasts have a statistically significant, positive association with realized volatility in equity markets in the 30 minutes after the speech (see the columns labeled *RV*). This holds true both for positive and negative news. Tail risk dynamics (see the columns labeled TR) are less strongly associated with central bank speech news signals in the high CPI regime.

**Low CPI regime**: A similar picture emerges in the low CPI regime. Speech-implied forecast revisions to CPI, good and bad, are strongly associated with increased equity market volatility. Low CPI regimes occur exclusively with normal or low GDP regimes (see Figure 6.4). Therefore, it is not surprising to see that speech-implied forecast revisions to GDP have a slightly stronger association with market volatility than during high CPI regimes, which almost exclusively co-occur with high GDP regimes. We interpret this as indicating that when the economy is in full swing, market sentiments tend to be optimistic and less 'attention' might be given to

central bank announcements. Tail risk in the low CPI regime seems to be sensitive to both positive and negative speech-implied forecast revisions to GDP.

**Normal CPI regime**: Normal CPI times are defined as periods when the inflation rate is close to 2%. During these periods, there are no longer statistically significant associations between speech-implied forecast revisions of any kind and market volatility, except for negative unemployment news. Again, we would interpret these results as indicating that markets 'listen' less attentively to central bank communication when the economy is in normal or good times compared to periods of undesirably high or low inflation. Table 6.11 shows similar patterns for the prediction of equity volatility and tail risk in the normal CPI regime.

### Bond Markets

Table 6.12 summarizes how speech-implied forecast revisions affect bond futures volatility across CPI regimes. Appendix C.6 details the regression tables for each CPI regime and target variable combination. Bond markets produce patterns similar to those in equity markets: large speech-implied forecast revisions are more significantly associated with higher bond volatility when CPI is far from the target.

Table 6.12: Association between speech-implied forecast revisions and volatility in bond markets across CPI regimes

| | **High CPI regime** | | | **Low CPI regime** | | | **Normal CPI regime** | | |
|---|---|---|---|---|---|---|---|---|---|
| | 2y | 5y | 10y | 2y | 5y | 10y | 2y | 5y | 10y |
| \|News CPI good\| | - | - | - | +*** | +*** | +*** | - | - | - |
| \|News CPI bad\| | - | +* | - | - | +* | +* | - | - | - |
| \|News GDP good\| | +* | - | - | +*** | - | - | - | - | - |
| \|News GDP bad\| | - | - | - | - | - | - | - | - | - |
| \|News U good\| | n/a | n/a | n/a | - | - | - | - | - | - |
| \|News U bad\| | +** | +** | +*** | - | +* | +* | +*** | +*** | +*** |
| n. obs. | | 33 | | | 42 | | | 52 | |

*+ = positive association. *= $p \leq 0.1$, **= $p \leq 0.05$, ***= $p \leq 0.01$. − = no statistically significant results. 'n/a' = no observations available.*

## 6.7.4 News Effects by GDP Regime

We now estimate equations (6.10) and (6.11) by different GDP regimes: low, normal, and high.

### Equity Markets

Table 6.13 reports speech-implied forecast revision effects on realized volatility and tail risk in equity markets, broken down by GDP regime. Appendix C.5 details these results for each CPI regime and target variable.

Table 6.13: Association between speech-implied forecast revisions and volatility in equity markets across GDP regimes

| | **High GDP regime** | | **Low GDP regime** | | **Normal GDP regime** | |
|---|---|---|---|---|---|---|
| | RV | TR | RV | TR | RV | TR |
| \|News CPI good\| | - | - | $+^{***}$ | - | - | $+^{**}$ |
| \|News CPI bad\| | - | - | $+^{***}$ | - | - | - |
| \|News GDP good\| | - | - | $+^{***}$ | $+^{***}$ | $+^{*}$ | - |
| \|News GDP bad\| | - | - | $+^{***}$ | $+^{*}$ | - | - |
| \|News U good\| | - | n/a | $+^{**}$ | $+^{**}$ | - | - |
| \|News U bad\| | $+^{**}$ | $+^{**}$ | $+^{***}$ | $+^{**}$ | - | - |
| n. obs. | 36 | | 44 | | 81 | |

*+ = positive association. *= $p \leq 0.1$, **= $p \leq 0.05$, ***= $p \leq 0.01$. − = no statistically significant results. 'n/a' = no observations available.*

**High and normal GDP regimes**: In high GDP times, negative speech-implied-forecast revisions to unemployment raise equity RV and TR. Similarly, positive speech-implied revisions to CPI forecasts raise TR during normal GDP periods.

**Low GDP regime**: In low GDP times, all speech-implied forecast revisions influence equity RV and all GDP and unemployment revisions influence equity TR. That is, RV and TR are a substantially more sensitive to forecast revisions during periods of low economic activity.

Overall, markets 'listen' most carefully in times of economic distress. In normal or good times, news in central bank speeches have less impact on RV and TR in equity markets.

### Bond Markets

Table 6.14 shows speech-implied forecast revision effects on realized volatility in bond futures markets, broken down by GDP regime. Appendix C.7 details these results for each GDP regime. Bond markets are also most sensitive to central bank speeches in extreme GDP regimes. Low GDP regimes witness the most significant association between GDP and unemployment forecast

revisions and bond volatility. But markets also appear to be more sensitive to central bank speeches in high GDP regimes than in periods of normal economic growth.

Table 6.14: Association between speech-implied forecast revisions and volatility in bond markets across GDP regimes

| | **High GDP regime** | | | **Low GDP regime** | | | **Normal GDP regime** | | |
|---|---|---|---|---|---|---|---|---|---|
| | 2y | 5y | 10y | 2y | 5y | 10y | 2y | 5y | 10y |
| \|News CPI good\| | $+^{***}$ | - | - | n/a | n/a | n/a | $+^{***}$ | - | - |
| \|News CPI bad\| | $+^{*}$ | $+^{*}$ | - | - | - | - | - | - | - |
| \|News GDP good\| | - | - | - | $+^{***}$ | $+^{***}$ | $+^{***}$ | - | - | - |
| \|News GDP bad\| | - | - | - | - | - | - | - | - | - |
| \|News U good\| | n/a | n/a | n/a | - | $+^{**}$ | $+^{**}$ | - | - | - |
| \|News U bad\| | $+^{*}$ | - | $+^{*}$ | $+^{***}$ | $+^{***}$ | $+^{***}$ | - | - | - |
| n. obs. | | 35 | | | 42 | | | 52 | |

*+ = positive association. *= $p \leq 0.1$, **= $p \leq 0.05$, ***= $p \leq 0.01$. − = no statistically significant results. 'n/a' = no observations available.*

## 6.8 Discussion

*Refer to updated and improved section in the published version in the Journal of Econometrics (December 2024)*

We use supervised multimodal natural language processing methods to map central bank language to forecasts of macroeconomic variables. We benchmark an extensive array of machine learning methods on this task. Finally, we apply this approach to a dataset of time-stamped speeches from Federal Reserve FOMC members in order to create an original monetary policy news series by taking the difference between central bank speech-implied forecast revisions and market expectations which we approximate with the latest available figures from the Survey of Professional Forecasters.

Our results indicate that news signals derived from central bank speeches can help explain volatility and tail risk in both equity and bond markets. Speech-implied news seem to carry information to which markets react - particularly in *abnormal* GDP and inflation regimes. We find no evidence that speeches resolve uncertainty. These findings underpin the importance of analysing the *continuous flow* of central bank communication with markets such as through FOMC member speeches.

Our analysis evaluates the market responses at the intradaily (high frequency) level. However,

speech-implied macroeconomic news may affect volatility and tail risk (and other financial market variables) differently over longer-term horizons. We plan to analyse the impact across different time horizons in future work. Equally, we aim to extend our work in follow-up explorations by analysing speech differences in characteristics and market effects between central banks and between central bank board members.

# 7

# Central Bank Communication and the Cacophony of Voices

This chapter applies the multimodal monetary policy analysis framework to estimate and analyse alignment in policy communication between central bank officials and develops a monetary policy news divergence index.

**Authors:** Maximilian Ahrens[1], Michael McMahon

**Publication venue:** EMNLP 2021, ECONLP Workshop [Ahrens and McMahon (2021)]

## 7.1 Abstract

Estimating the effects of monetary policy is one of the fundamental research questions in monetary economics. Many economies have been facing prolonged periods of ultra-low interest rate environments ever since the global financial crisis of 2007-08. The Covid pandemic recently added another such ultra-low interest rate regime. During those periods, in the US and Europe, interest rates were close to (or even below) zero, which limits the scope of traditional monetary policy measures for central banks. Dedicated central bank communication has hence become an increasingly important tool to steer and control market expectations these days. However, incorporating central bank language directly as features into economic models is still a very nascent research area. In particular, the content and effect of central bank speeches has been mostly neglected from monetary policy modelling so far. With our paper, we aim to provide to the research community a novel, monetary policy shock series based on central bank speeches. We use a supervised topic modeling approach that can deal with text as well as numeric covariates to estimate a monetary policy signal dispersion index along three key economic dimensions:

[1] main author

GDP, CPI and unemployment. This *dispersion shock* series is not only more frequent than series that classically focus on policy announcement dates, it also opens up the possibility of answering new questions that have up until now been difficult to analyse. For example, do markets form different expectations when facing a 'cacophony of policy voices'? Our initial findings for the US point towards the fact that more dispersed or incongruent monetary policy stance communication in the build up to Federal Open Market Committee (FOMC) meetings is associated with stronger subsequent market surprises at FOMC policy announcement time.

## 7.2 Introduction

Understanding the (causal) effect of monetary policy on economic and financial variables is one of the most fundamental empirical questions in monetary economics. An extensive branch of literature focuses on this question; a key empirical reference is Christiano et al. (2005). In this theme, we address this pivotal monetary economics question, and in particular, the effect of central bank communication. Our research question is whether we can extract and measure policy signals, uncertainty, or shocks from central bank speeches.

**Monetary policy shock identification:** The starting point of any attempt to understand the causal macroeconomic effects of monetary policy is to identify the unanticipated, exogenous element in monetary actions to avoid endogeneity concerns. The literature on constructing these monetary shocks is vast. Some use VAR analysis (Christiano et al., 2005), others focus on the narrative approach (Romer and Romer, 2004), and, most recently, the emphasis has been on high-frequency identification (Gürkaynak et al., 2005; Gertler and Karadi, 2015; Nakamura and Steinsson, 2018; Gertler and Horvath, 2018). These are not all the same thing. Monetary shocks from the narrative approach, such as the Romer and Romer (2004) shocks, are exogenous to macroeconomic conditions but are not immediately observed outside the Federal Open Market Committee (FOMC); high-frequency surprise measures, such as in Gertler and Karadi (2015) represent the surprise for markets but may be endogenous to macroeconomics. As Ramey (2016) stresses, the focus on deviations from the systematic response of policy, or shocks, is "a search for instruments rather than for primitive macroeconomic shocks". However, essentially all approaches use changes in actual policy rates as part of the identification strategy. Empirically, though, there is very little surprise in monetary policy announcements: the average change in the Fed Fund futures market around announcements since the 1990s is only 2.7 basis points. To the

extent that central bank communication between meetings shapes expectations of subsequent policy decisions, it contains the true surprises. What is more, a pure focus on central bank meetings only provides a rather infrequent updating cycle.

**NLP-based central bank speech analysis:** In contrast to the 6-8 week frequency of FOMC meeting announcements, central bankers give speeches frequently throughout inter-meeting periods, oftentimes commenting on the economic and financial conditions as well as on monetary policy considerations. We therefore create a novel series of monetary policy shocks based on central bankers' public speeches; our key innovation is to use supervised natural language modelling approaches to map speeches into implied exogenous policy shocks. These implied shocks can then be used as instruments to identify the impact of monetary policy decisions on economic outcomes, or to explore the interaction of monetary transmission with communication. The main challenge to using inter-meeting communication is empirical; we need to map speeches, which are text documents, into implied policy signals.

**Motivation - economic modelling approach:** Our empirical strategy is somewhat motivated by the seminal work of the monetary policy shock series by Romer and Romer (2004), who use the residual of a regression of the change in the Fed Funds Rate in each FOMC meeting on forecasts of future economic conditions as measured by the Greenbooks. The Greenbooks do not only contain numeric forecasts, as used by Romer and Romer, they also contain extensive text sections explaining and describing the respective forecasting exercises. We pair the Greenbook text sections with their respective numeric forecasts - focusing on three key dimensions: GDP, inflation (CPI), and unemployment. Using supervised text representation learning that jointly considers both text features as well as numeric features, we obtain text representations that map from Greenbook language to forecasts. This is where our setup substantially differs from the Romer and Romer approach or from a pure prediction exercise. We try to learn domain specific text features that have strong explanatory power over the target variables (the Greenbook forecasts), whilst controlling for the influence of other numeric covariates. The assumption here is that if there is any economic meaning captured in those text representations, they should be transferable to other related datasets. In our case, this would be the central bank speeches, for which we have the speeches' text but obviously no Greenbook forecast figures, as their next update will only be disclosed at the next FOMC meeting. For each speech, we can then estimate

a) the implied monetary policy signal on changes in GDP, CPI and unemployment forecasts and b) establish a measurement for the information dispersion across central bank speeches, which is an interesting economic measure in itself when it comes to assessing the effectiveness of central bank communication and guidance and its effect on market expectations.

**Motivation - topic modelling approach:** To construct the mapping from central bank texts to forecasts, we use supervised learning methods in form of topic models that can incorporate both numeric covariates as well as labels (Card et al., 2018; Ahrens et al., 2021). The key reasons why we opted for a topic modelling approach to represent the text features are that such models yield a reasonably high level of interpretability whilst working reliably even in research settings with relatively 'small' datasets compared to more mainstream NLP applications with millions of data points. Supervised topic models, such as Card et al. (2018); Ahrens et al. (2021), allow us to learn the domain specific text representation whilst controlling for other numeric covariates such as macroeconomic and financial market conditions, which also potentially affect the textual content of central bank reports. The use of generative topic models can further be motivated from the economic modelling side. Generative models are akin to structural models in economics, and provide a complete description of the joint distribution of text, covariates, and dependent variables (policy signals in our case).

**Our contribution:** With this paper, we aim to provide to the research community a novel, monetary policy shock series based on central bank speeches.[2] We construct a monetary policy signal dispersion index along three key economic dimensions: GDP, CPI and unemployment. This shock series is not only more frequent than series that focus on FOMC meetings, it also opens up the possibility of answering new questions that have up until now been difficult to analyse. For example, do markets form different expectations when facing a 'cacophony of policy voices', i.e. less aligned policy communication? Our initial estimates suggest there might be evidence for it. Finally, we further advance the empirical use of machine learning and data-science methodologies in economics.

[2]available at: github.com/MaximilianAhrens/data/tree/main/central_bank_speech_signals

## 7.3 Related Work

### 7.3.1 Economics - Monetary Policy and Central Bank Communication

In the introduction, we already outlined the different mainstream approaches on identifying monetary policy shocks (Romer and Romer, 2004; Gürkaynak et al., 2005; Gertler and Karadi, 2015; Nakamura and Steinsson, 2018; Gertler and Horvath, 2018). The monetary policy literature suggests there is evidence that central bank communication can have an impact on an array of different financial market instruments, for example see Gürkaynak et al. (2005); Boukus and Rosenberg (2006); Blinder et al. (2008); Carvalho et al. (2016). However, whilst some papers on monetary policy shocks consider the timings of central bank meetings and announcements, they tend not to look into the actual language content of the central bank communication. With the onset of more accessible natural language processing models for the wider research community, this has recently started to change. Bholat et al. (2015) introduced initial text mining and language modelling approaches for central bank communication. Shiller (2017) brought forward the notion of *Narrative Economics*, suggesting the importance of language-based narrative in forming public beliefs and emphasizing more systematic incorporation of information conveyed through language into economic modelling. For monetary policy in particular, Haldane and McMahon (2018) outline the importance of central banks' roles in shaping public narrative on economic conditions and uncertainties. Hansen and McMahon (2016) use dictionary methods and topic models analysing the content of central banks' forward guidance and find that it has larger effects on financial markets than announced views of current economic conditions. Ahrens (2018) extends the Romer and Romer (2004) shock series with topic features based on FED Beigebook to extract a more exogenously driven monetary policy shock series that reconciles recent empirical data with monetary policy theory. Ochs (2021) builds on this text analysis framework for monetary policy shocks as well and comes to similar conclusions. Hansen et al. (2019) analyse the Bank of England's Inflation Reports via topic modelling and find that communication plays an important role in shaping perceptions of uncertainty in long-run interest rates.

Some former monetary policy makers believe that monetary policy decisions hold greater

weight with markets when the committee communicates a single message (Schonhardt-Bailey, 2013) . The open question is how should a central bank "communicate effectively and honestly" (Blinder, 2018) when the central bank has multiple decision makers who, naturally given the complex nature and uncertainty of the decisions, often disagree. While many worry about the effects of a cacophony of voices (i.e. the misalignment in policy communication), there is no hard evidence on the exact extent or nature of it in practice. Compared with using the policy shocks associated with the announcements, our shock series is uniquely placed to address this cacophony of voices problem. Firstly, the cacophony arises most generally through speeches and interviews (individual member communications) rather than through the statements or even minutes; while the existing literature typically ignores the signals in individual member communications, they are the focus of our analysis. Yellen (2017) suggests that one of the main disruptive effects of policymakers' public speeches is the transmission of disagreement regarding individual short-run policy goals. Our shocks capture exactly this. We can use their range and variance as a measure of cacophony.

### 7.3.2 NLP - Modelling with Numeric and Text Data

We use a supervised topic modelling approach that learns a domain specific text representation that is optimized to predict the target variable together with other numeric (also referred to as tabular) covariates.

Topic models are a popular choice when it comes to incorporating text features into, for example, social and data science models (Gentzkow et al., 2019). Many topic models have built on the seminal work by Blei et al. (2003). Supervised topic models such as Blei and McAuliffe (2008); Zhu et al. (2012); Chen et al. (2015) allow to infer topics that are relevant for predicting a domain specific label. Topic models such as Eisenstein et al. (2011) and Roberts et al. (2014) take into account the effect of numeric covariates on the topic distributions but do not explicitly use labels to guide the topic discovery process. However, recent supervised topic models such as Card et al. (2018); Magnusson et al. (2020); Ahrens et al. (2021), have combined those previous two approaches, which allow for jointly learning text representations and prediction parameters based on both labels and numeric covariates. Such topic model class is the most suitable for our research setup, where our labels are the numeric forecasts in the Greenbook section, which we

want to predict based on their associated text sections as well as other relevant economic and financial indicators at that time. Card et al. (2018) have proposed such model (SCHOLAR) for classification tasks. Ahrens et al. (2021) propose a Gibbs-sampled alternative to SCHOLAR, BTR, as well as a regression extension of SCHOLAR, rSCHOLAR.[3] The research task here is prediction and not causal inference. We therefore opt for rSCHOLAR instead of BTR. As shown in chapter 5, both models demonstrate comparable predictive performance. However, rSCHOLAR is more computationally efficient as it is solved via variational inference instead of Gibbs sampling.

## 7.4 Data

Our empirical dataset consists of two distinct yet related data subsets - the FED's *Greenbook* data and the *public speeches* of its central bankers. The first one comprises all numeric and text data captured in the 145 Greenbooks released from 1990 to 2013. [4] The Fed drafts a new Greenbook report about every 6-8 weeks in the run-up to the FOMC meetings. The Greenbook data contains numeric estimates on a multitude of contemporary economic figures as well as forecasts for several time horizons. Those numbers are accompanied by paragraphs which, for instance, put the forecasts into context and explain the rationale behind them. Different sections in the Greenbook focus on different economic and financial indicators. We use the provided separation by headlines and sections to obtain a granular mapping about which text passages pertain to which numeric figures. Our second data subset comprises textual transcripts of over 3000 speeches given by different central bank officials over the time-span from 1993-2013.

## 7.5 Topic Model

SCHOLAR (Card et al., 2018) is a supervised topic model that generalises both sLDA (Blei and McAuliffe, 2008) as it allows for predicting labels, and SAGE (Eisenstein et al., 2011) which handles jointly modelling covariates via 'factorising' its topic-word distributions into deviations from the background log-frequency of words and deviations based on covariates. SCHOLAR is solved via neural variational inference (Kingma and Welling, 2014; Rezende et al., 2014). However, it was not primarily designed for regression tasks. We therefore use rSCHOLAR, an extension by Ahrens et al. (2021), which incorporates linear and non-linear regression layer

[3] https://github.com/MaximilianAhrens/scholar4regression
[4] Greenbook data is released to the public with a 5 year delay

options in the prediction network of the model. As we don't further modify this model, we outline the generative process here and refer to Card et al. (2018) for more details on the model: **for** each document $i = 1, \ldots, D$:

1. $\boldsymbol{\zeta}_i \sim \mathcal{N}\left(\boldsymbol{\zeta} | \boldsymbol{\mu}_0(\alpha), \text{diag}(\boldsymbol{\sigma}_0^2(\alpha))\right)$
2. $\boldsymbol{\theta}_i = \text{softmax}(\boldsymbol{\zeta}_i)$
3. $\boldsymbol{\eta}_i = f_{\text{gen}}(\boldsymbol{\theta}_i, \boldsymbol{c}_i)$
4. **for** each word $n = 1, \ldots, N_d$ in document $i$:
   (a) $w_{i,n} \sim \text{Multi}(w | \text{softmax}(\boldsymbol{\eta}_i))$
5. $\boldsymbol{y}_i \sim p(\boldsymbol{y} | f_y(\boldsymbol{\theta}_i, \boldsymbol{c}_i))$

where $\boldsymbol{\zeta}$ is the reparametrisation variable (Kingma and Welling, 2014), and $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ are the mean and diagonal-variance parameters of the logistic normal prior for document-topic distribution $\boldsymbol{\theta}$ (Srivastava and Sutton, 2017). $\alpha$ is a Dirichlet hyperparameter for $\boldsymbol{\zeta}$. $\boldsymbol{\eta}$ are the topic assignments, $\boldsymbol{c}$ are the numeric covariates, $w$ the words and $\boldsymbol{y}$ the label (or target variable). $f_{\text{gen}}$ is a neural network for the generative topic modelling part. $f_y$ is the prediction layer part of the model, which can be chosen to be virtually any form of adequate neural network structure. In our case, we focus on i) the special case where it is just a linear regression and ii) when topics and covariates are allowed to interact with each other through feed-forward layers. Regression network $f_y$ and generative network $f_{\text{gen}}$ are jointly optimized via backpropagation using Adam (Kingma and Ba, 2015).

## 7.6 Economic NLP Model

As described earlier, the model estimation process is broken down into two stages: 1) learning the mapping from central bank language to economic conditions, 2) applying the learned mapping to central bank speeches. This section will outline the estimation equations at each stage and for the three economic signals: GDP, CPI, and unemployment.

### 7.6.1 Stage 1 - Learn Mapping from Central Bank Language to Economic Conditions

In the first stage, we learn text representations that map from the FED's Greenbook texts to its forecasts. We categorize the Greenbook sections according to which forecast they pertain to. Subsequently, we estimate a separate mapping equation for each of the three distinct economic signals. For each of these equations respectively, the left hand side is the FED's Greenbook forecast for GDP, CPI or unemployment over the next year. We control for both the latest contemporary values of GDP, CPI and unemployment as well as the forecast values in the previous Greenbook report. The respective Greenbook text sections serve as the text features for which we want to learn their association with the corresponding Greenbook forecasts, controlling for the influence of the numeric covariates.

As an example, we show below the mapping equation for CPI, where the target variable is denoted as $\Delta\phi_{4:0,m}$. It represents the change in the CPI forecast $\pi$ over the next year at FOMC meeting timestamp $m$. The target variable for GDP is $\Delta g_{4:0,m}$ and $\Delta u_{4:0,m}$ for unemployment. Otherwise, the equations for GDP and unemployment have the same RHS variables except that the text corpus is each time specific for the respective Greenbook forecast section, i.e. $\theta_\pi$ represents the topic mixtures for the CPI corpus. Similarly we have $\theta_g$ for GDP and $\theta_u$ for unemployment. $\theta_{\{\pi,g,u\},k}$ represents the $k^{th}$ topic feature for the respective corpus. In the linear case, we can write out the entire explicit regression equation for $\boldsymbol{y}_i \sim p(\boldsymbol{y}|f_y(\boldsymbol{\theta}_i, \boldsymbol{c}_i))$ from above quite easily as

$$\begin{aligned}\Delta\phi_{4:0,m} &= \rho_u u_{0,m-1} + \rho_\pi \pi_{0,m-1} + \rho_g g_{0,m-1} \\ &+ \rho_{\Delta u}\Delta u_{4:0,m-1} + \rho_{\Delta\pi}\Delta\pi_{4:0,m-1} \\ &+ \rho_{\Delta g}\Delta g_{4:0,m-1} + \sum_{k=1}^{K}\omega_k\theta_{\phi,k} + \epsilon_m, \end{aligned} \tag{7.1}$$

where $\rho$s and $\omega$s represent the regression weights and $\epsilon$ is the measurement error. If we thought in the spirit of the narrative approach in Romer and Romer (2004), we could now divide the RHS of this equation into two economically meaningful parts - the 'policy preference' component and the 'policy shock' component. The regression parameters $\omega$ serve as the policy mapping from numeric $(u, \pi, g)$ and text $(\boldsymbol{\theta})$ data features to the related central bank forecast. They

represent the estimated 'policy preference' function of the equation. Under the assumption that the numeric and text features cover the relevant information space for the central bank to form its forecasts, the regression residual can be seen as the part of the forecast or policy decision that cannot be explained by the information accessible to the central bankers and therefore would be considered as some sort of an exogenous monetary policy shock to an observer. Our primary focus lies on identifying the policy mapping function rather than the classical Romer and Romer policy shock component. We want to identify this mapping and then subsequently apply it to central bank speeches. Equation (7.1) is being estimated with rSCHOLAR, which jointly estimates the topic mixtures and regression parameters in order to best explain the target variable.

### 7.6.2 Stage 2 - Apply Mapping to Central Bank Speeches

In the second stage, we take the estimated mapping from stage 1 and apply it to our central bank speeches dataset. We take the estimated regression parameters from stage 1, $\hat{\rho}$ and $\hat{\omega}$ as well as the estimated domain specific topic features $\hat{\boldsymbol{\theta}}$ and apply it to each speech. The numeric features will be the last Greenbook forecasts that a central banker will have had access to at a given point in time $(m-1)$, which is in line with the regression setup in equation (7.1). We then obtain (i) an implied monetary policy signal for the respective target variable, and (ii) a measure of signal dispersion by assessing the range of signals that central bankers convey during any inter-meeting period.

## 7.7 Results

The results section is divided into three main steps that build upon each other. In the first step, we assess the quality of our mapping from Greenbook texts to forecasts and the implied predictive signal for the different policy dimensions that we estimate out of sample in the speeches. In the next step, we then derive policy signal dispersion measures from those implied predictive signals. Finally, we use the calculated dispersion indices to estimate some initial market and policy transmission effects.

### 7.7.1 Estimating Implied Signals in Speeches

In the first stage of our model, we aim to identify meaningful text representation in the Greenbook dataset. The training has been done by splitting the Greenbook dataset into a training and a validation set.[5] We trained the models for 2000 epochs, however the validation set optimum was virtually always achieved significantly earlier. The K=20 model with interaction terms yielded the best validation set results across topic size (range 3-50 topics[6]) when evaluated according to minimizing mean-squared error in the regression part and minimizing perplexity in the topic modelling part (topic compositions shown in Appendix D.1, validation set performance Appendix D.2). Table 7.1 shows the training set (on Greenbook data) and test set (on speech data) predictive $R^2$, which is defined as $R^2 = 1 - \frac{mse(data)}{var(data)}$. It reflects the percentage of explained variance of the respective target variable. The bottom part of the table shows that the model with numeric and text features fits the Greenbook validation /training data a lot more accurately than the purely numeric baseline model. Those results might be an indicator that our NLP model learned relevant text representations. However, purely judging on the training results does not yet give us much insight into whether these results might be a mere artefact of potential overfitting. When looking at the upper half of Table 7.1, we get assurance that our model did not just fit noise in the training data. The supervised mapping from Greenbook language to forecasts has been applied to the speeches dataset which had not been used at all as part of the training process. The NLP model substantially outperforms the purely numeric model. It explains 66% more out-of-sample variance in the speech data for the CPI target, 10% more variance on the GDP target, and 8% more variance on the unemployment target. All results are based on mean outcomes over 50 model runs per target. Figure 7.1 visualizes the actual FED forecast series (blue) and the predicted forecast values based on the speech dataset.

### 7.7.2 Estimating Speech Dispersion

We can now construct economic signal dispersion measures, based on the estimated implied signals on future FED forecast changes on GDP, CPI and unemployment, which we derived from the speeches. For each of the three dimensions, we calculate the range of the implied signal for

[5] Training-validation randomly split 80-20

[6] larger topic models ($K > 50$) did not yield better MSE results and were therefore dropped from further analyses.

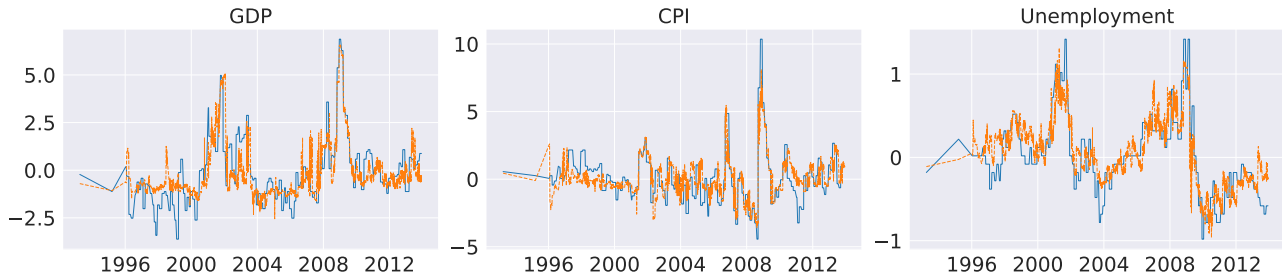



Figure 7.1: Out of sample implied policy signals

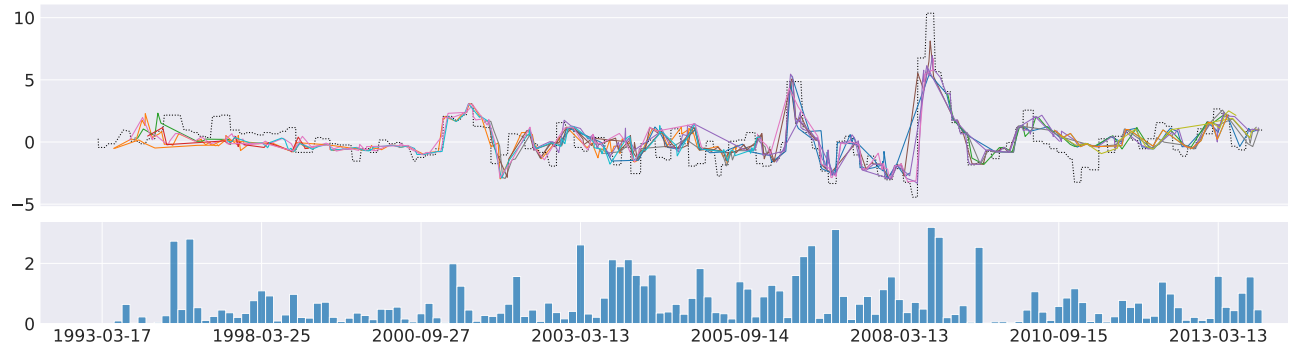
*Realised value (blue), model prediction (orange)*

| predictive $R^2$ | numeric | numeric + text |
|---|---|---|
| Speeches - GDP signal | 0.524 | **0.577 (0.016)** |
| Speeches - CPI signal | 0.346 | **0.575 (0.039)** |
| Speeches - Unempl. signal | 0.630 | **0.681 (0.019)** |
| Greenbook - GDP training | 0.502 | **0.766 (0.080)** |
| Greenbook - CPI training | 0.295 | **0.790 (0.147)** |
| Greenbook - Unempl. training | 0.458 | **0.657 (0.011)** |

Table 7.1: Predictive $R^2$. Models trained on Greenbook dataset, tested on speeches dataset. Best model in bold. Reported means across 50 model runs, standard errors in brackets. Numeric (OLS) has analytical solution.

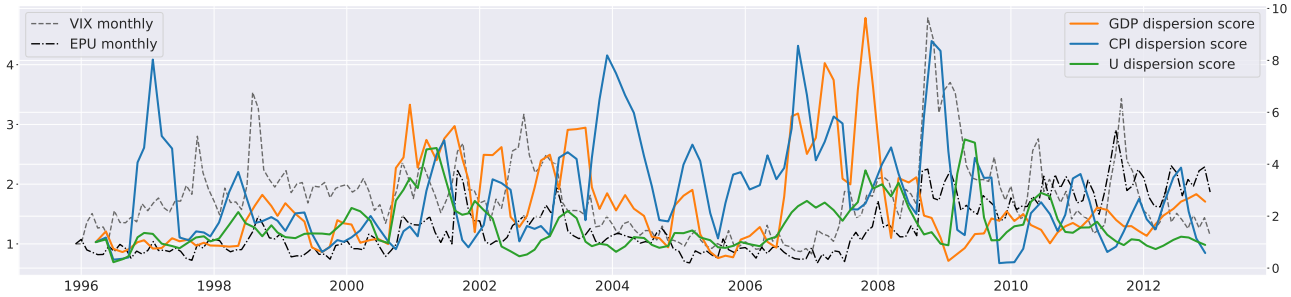
each inter-meeting period of the FOMC. This is simply done by taking the distance between the most positive and most negative speech signal. There are further ways to refine this measure that we are currently exploring, for instance, by weighting the signal according to an author-relevance metric. The FED chairperson's speeches might intuitively carry higher weight than other FED officials' announcements. Figure 7.2 visualizes all inferred CPI signals across speeches of all central bankers who spoke during the period of our data sample. Equivalent dispersion figures for GDP and unemployment are in Appendix D.3. Intuitively, higher dispersion in the signal conveyed by central bank officials could be perceived as a less united policy stance and therefore a higher degree of uncertainty about future policy guidance and decision making. In a next step, we compare our policy signal dispersion measures against two common market and policy uncertainty measures - the VIX and the Economic Policy Uncertainty (EPU) index (Baker et al., 2016). Figure 7.3 shows how these indices compare over time. Our dispersion indices tend to increase in similar periods when also VIX and the EPU indicate higher market uncertainty. Furthermore, our dispersion measures seem to provide a more granular insight into specific monetary policy uncertainties. As an example, before and at the onset of the global financial

Figure 7.2: Out of sample estimation of monetary policy signals on CPI




*Top figure: signal by individual central banker speaker. Bottom figure: derived dispersion measure (grouping window: inter-FOMC-meeting periods).*

Figure 7.3: Dispersion scores for GDP, CPI and unemployment compared to VIX and EPU index




*All indices re-indexed to beginning of displayed time-series.*

crisis in 2007-9, there seems to have been a relatively high degree of dispersion in terms of the CPI and GDP stance. During the crisis however, the FED appears to have communicated with a much more united and aligned voice. In the aftermath of the crisis then, as some pundits feared a threat of inflationary pressures due to ultra-low interest rates, the CPI dispersion score jumps up whilst the signals for GDP and unemployment remain more united.

If we go back and look into the raw texts of the speeches at the top and bottom end of this CPI dispersion peak around end of 2008, we find that the speech that signalled the strongest "likelihood for higher future inflation/more dovish monetary policy stance on inflation" was given on 19th of November 2008 by Donald Kohn [7] who is generally considered a moderate dove in terms of monetary policy. In this speech, he quite clearly expresses his inflation policy view (full transcript of speech in Appendix D.4). He sums up his speech with the words:

*"[... ]In sum, I am not convinced that the events of the past few years and the current crisis demonstrate that central banks should switch to trying to check speculative activity through tighter*

[7] Former Vice Chair of the Federal Reserve.

*monetary policy whenever they perceive a bubble forming. [...] For these reasons, the case for extra action still remains questionable, despite our having learned that the aftermath of a bubble can be far more painful than we imagined.[...] ".*

On the flip side, one of the speeches perceived as the most "hawkish" during the same time window was given by Jeffrey M. Lacker [8], where he expressed his policy stance on inflation in his speech on 3rd of December 2008 with:

*"[...]Since 2004, overall inflation has trended upward, and has been higher than I would like, over the last few years. [...]Many economists are forecasting relatively low inflation in the months ahead, on the grounds that widening economic slack is generally associated with declining price pressures. [...]I would be cautious about relying on it as a causal relationship. And while it may seem premature to be worrying about how inflation behaves after the recession is over, we need to be sure our policy remains consistent with a strategy that does not allow inflation to ratchet up over the business cycle.[...]."*

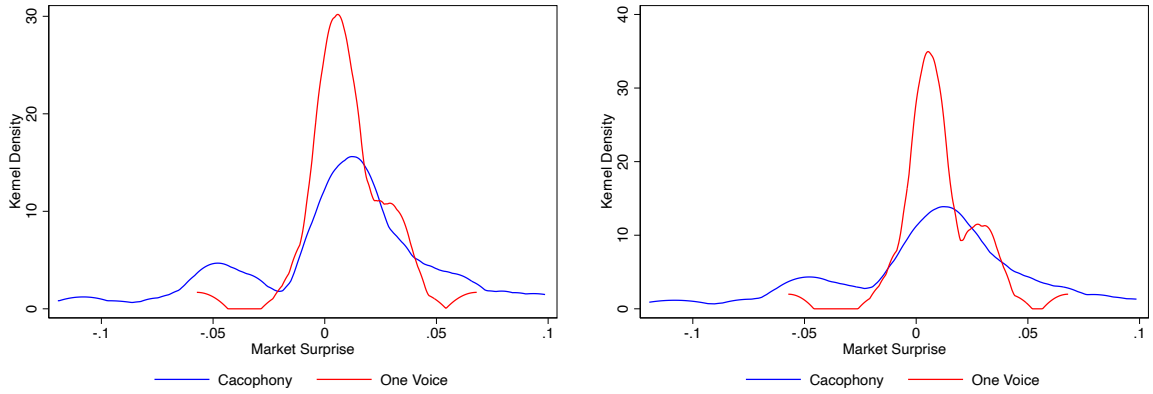
### 7.7.3 Estimating Dispersion Effects

In order to examine the effect of FOMC members singing, or not, from the same hymn sheet, we use the three dispersion indices to create a variable indicating inter-meeting periods as characterised either by Cacophony or One Voice. There are many ways to define such an indicator but for simplicity, and to protect against outliers which would affect the standard deviation or range of the signal distribution, we use the interquartile range (IQR) for each of the GDP, CPI and unemployment series. We then average these dispersion series to get a single composite dispersion index; as an alternative, we average the two most-dispersed series, recognising that cacophony could be driven by different signals on only a subset of the three indicators. We then define a period as one of *Cacophony* when the series is above the median, and *One Voice* when below.

Figure 7.4 plots the Kernel Density of market surprises at the FOMC policy announcement after the inter-meeting period in which the speeches are measured. These market surprises are calculated using a narrow, 30-minute window around the FOMC announcement and, therefore, should reflect all market information priced into asset prices right before the meeting. The left figure is constructed using the average of all three signal series, and the right figure is based on

[8]Former President, Federal Reserve Bank of Richmond

Figure 7.4: Kernel Density of Market Surprises




*Kernel density of market surprises when preceded by cacophony (blue), and One Voice (red). We implement two different measures of the Cacophony Indicator - the left figure shows a measure based on above median values of the average of all three signal series and the right figure looks at just the two most dispersed signal series each period.*

the series using the two most dispersed signal series each period. Our estimates suggest that the periods in the run-up to FOMC meetings that carried a higher degree of cacophony, are associated with larger market surprises at FOMC announcement time.

More formally, we run a simple OLS specification to see if our dispersion series can predict the market news. The specification involves regressing the absolute value of the market surprise on a number of controls, including an NBER recession indicator, the count of the number of speeches, as well as measures of market volatility (VIX) and uncertainty (BBD). The average signal for each indicator is also included. Table 7.2 reports the main findings using our two composite dispersion indices described above.

Cacophonous periods predict market surprises even when controlling for economic conditions, suggesting that our policy dispersion signals capture an important dimension of central bank communication.

## 7.8 Discussion

With our paper, we aim to provide to the research community a novel, monetary policy shock series based on central bank speeches. Based on a supervised topic modelling approach, we construct a monetary policy signal dispersion index along three key economic dimensions: GDP, CPI and unemployment. This *dispersion shock* series is not only more frequent than series that focus purely on FOMC meetings, it also opens up the possibility of answering new questions that

Table 7.2: Estimates of the effect of Cacophony on subsequent market surprise (Mkt News)

| Regressors | (1) Mkt News | (2) Mkt News | (3) Mkt News |
|---|---|---|---|
| Lagged Dispersion Index | | 0.019*** | |
| | | [0.002] | |
| Lagged Dispersion Index (alt) | | | 0.014*** |
| | | | [0.004] |
| $R^2$ | 0.180 | 0.262 | 0.261 |

*Higher dispersion index implies more cacophony. All regressions include control variables: NBER recession indicator, count of number of speeches, measures of market volatility (VIX), and uncertainty (BBD).*

have up until now been difficult to analyse. For example, is monetary policy communication more effective when communicated with 'one united voice' to the markets and likewise do markets form different expectations when facing a 'cacophony of policy voices'. Our initial findings point towards the fact that more 'cacophonous' policy communication in the build-up to FOMC meetings might be associated with stronger subsequent market surprises at FOMC policy announcement time. With our work, we hope to encourage and facilitate further research in this area.

# 8 Conclusion and Future Work

## 8.1 Conclusion

In this DPhil thesis, we explored the use of natural language processing (NLP) methods for economic and financial modelling. We did so with four contributions:

**In Chapter 4, we introduced foundational work towards a systematic NLP benchmark for economics and finance**. Over the past years, researchers introduced various NLP methods in these domains in order to answer new research questions. It is time to take stock and start systematically evaluating which NLP methods work best for the most common domain specific tasks, given the most common domain specific dataset characteristics. In particular, we want to underline the importance to evaluate multimodal (for now, text and numeric data) tasks, since many text datasets in economics and finance come accompanied by potentially relevant numeric metadata. The key findings of our initial benchmarking evaluation are:

1. Financial transformer models perform best on text-only classification datasets.

2. Financial transformer models struggle on multimodal economics and finance datasets, highlighting more need for concerted research efforts on fusing multimodal data and pre-training/fine-tuning models for tasks in these domains. Current financial transformers, such as FinBERT (Araci, 2019) and FLANG-BERT. (Shah et al., 2022) can also only sequence 512 tokens at a time. A quite substantial limitation given many datasets in finance and economics have much longer average sequence lengths.

3. The Loughran and Mcdonald (2011) dictionary method - still widely used as a default text analysis tool in economics and finance - underperforms substantially across all evaluated

datasets.

4. Relatively simple word count models do very well across all datasets.

**In Chapter 5, we introduced the Bayesian Topc Regression (BTR) model.** BTR is a multimodal NLP algorithm based on a supervised topic model that can jointly model text and numeric data. We introduced this model based on the need for better multimodal NLP methods particularly for the domains of economics and finance. We also showed that our BTR model respects the Frisch-Waugh-Lovell theorem - an important condition to make progress towards causal inference with observational text data. Causal inference with text data is a rather nascent research field which has seen important contributions over the past years. With the BTR model, we are providing the research community with a method for more reliable causal inference with text data, in an causal identification framework commonly used in the economics and finance literature. Our model also demonstrated to be competitive in prediction tasks that rely on multimodal data containing both text and numeric data.

**In Chapter 6, we then developed a multimodal NLP modelling framework for the application space of monetary policy communication analysis.** We proposed an identification framework for monetary policy shocks in central bank communication such as speeches, to map these speeches to forecasts of macroeconomic variables. As part of our work, we constructed a novel dataset of speeches from Federal Reserve FOMC members. We then tested the framework on an extensive array of machine learning models that can fuse text and numeric data. Our empirical results are promising, as our best performing multimodal NLP model has an non-negligible predictive edge over purely numeric models when it comes to forecasting SPF revisions based on central bank speeches. These results lent empirical evidence to our aim of learning a generalisable mapping from central bank language to forecasts. We then created a novel monetary policy news series by taking the difference between our speech-implied forecasts and the latest available figures from the Survey of Professional Forecasters. Our results indicate that news signals derived from central bankers' speeches can help explain equity and bond market volatility as well as tail risk. This finding underpins the importance of analysing the continuous communication process of central banks with the markets – and speeches are

a main communication element in this. Speech-implied news signals seem to carry relevant information to which markets react. We find no evidence that implied speech signals resolve uncertainty, at least not in the short run.

**Finally, in Chapter 7, we use the introduced identification framework for monetary policy news shocks to analyse central bank communication alignment.** Based on our multimodal NLP framework, we constructed a monetary policy signal dispersion index along three key economic dimensions: GDP, CPI, and unemployment. We derived a *news dispersion* series that is not only more frequent than series that focus purely on FOMC meetings, but it also opens up the possibility of answering new questions that have up until now been difficult to analyse. For example, is monetary policy communication more effective when communicated with 'one united voice' to the markets and likewise do markets form different expectations when facing a 'cacophony of policy voices'? Our findings point towards the fact that more cacophonous policy communication in the build-up to FOMC meetings might be associated with stronger subsequent market surprises at FOMC policy announcement time.

Overall, we hope to have contributed to a better understanding of how NLP methods can be leveraged for economic and financial modelling – by providing an NLP benchmark foundation for economics and finance, by having contributed to the field of multimodal NLP modelling in economics and finance, and by having introduced a new empirical identification framework for monetary policy shocks. As our work has also shown in many parts, more concerted research is needed to better integrate NLP models into economic and financial modelling frameworks. We outline below a few general perspectives on NLP for economics and finance and key areas for future work.

## 8.2 Future Work

**NLP benchmark:** For the NLP benchmark work, we have several extensions in mind. After all, this project is designed to establish a foundation to which the research community can continuously add. In particular, we plan to extend our benchmark model suite by including long-document transformers and open-source equivalents of GPT-3 and GPT-4. Furthermore, we aim to extend our benchmark task suit beyond sentiment analyses and sequence classification.

Additional tasks we aim to include are for example named-entity-recognition and question answering. Finally, we aim to make our benchmark more interactive, setting up a website that the research community can easily use to add new models or even new datasets to the benchmark.

**NLP for monetary policy analysis:** We aim to further refine our empirical identification framework, to identify more nuanced signals in central bank communication. From the methodological side, one research extension is to develop and leverage more advanced NLP models to extract more precise and nuanced economic signals. More details on this are provided in the next paragraphs. From the empirical monetary policy analysis side, we also aim to explore additional research questions. For example, we have so far not distinguished between the news impact of different FOMC members. To what extent might the chair's voice carry more weight than that of junior members? Equally, so far we have solely analysed the US jurisdiction. Extensions could cover other major economic areas such as the UK, the Eurozone, Canada, and Japan.

**Narrative and symbolic reasoning:** We have been awarded a research grant by the Alan Turing Institute to conduct research on multimodal language and graph models for narrative detection and analysis. This work is not covered in this DPhil thesis. However, representing narrative – or other key informational elements such as factual relations – in symbolic form and instilling it into language models, might be another promising future path towards more context-aware language models, reduced model hallucinations, increased factuality, and overall increased model performances.

**Multimodal models:** We aim to continue working on multimodal NLP models for economics and finance. As our findings have shown, the fusion task of text and numeric data, particularly with transformer models in these domains is far from being conclusively solved. Yet, recent advances in classic NLP tasks show us what potential could be unlocked if we start to better integrate transformer algorithms into our economic and financial modelling frameworks. One pathway can be the development and exploration of multimodal, long document, financial transformers. Furthermore, in initial ad-hoc experiments with chatGPT and other prompt-based

LLMs, we found that such models performed rather well in extracting key informational bits on inflation, GDP, or unemployment given an economic text and additional context information about the contemporaneous economic conditions. The integration of the latest (prompt-based) large language models such as chatGPT (based on GPT-3.5 or GPT-4) or similar open-source models might be the first step towards a new era of integrating NLP methods in economic and financial modelling frameworks.

## 8.3 Perspectives on NLP for Economics and Finance

**Choosing the right NLP model:** This thesis has demonstrated that oftentimes relatively small and specialised models can still achieve competitive results compared to large language models. Especially when dealing with dataset characteristics that are common in macroeconomics and finance such as (1) relatively few data points (thousands rather than millions), (2) relatively long text lengths (thousands of tokens per document), and (3) multimodality (text and numeric data), large language models do not always necessarily have the upper hand – at least not yet. However, large language models are becoming increasingly more powerful – for example as they can handle multimodality increasingly better and as their context window size constantly increases. Simultaneously, the computational inference costs of such large language models keeps on decreasing as we make further research progress. It will likely be a wise research strategy to run an experiment with smaller and simpler NLP models initially in order to obtain a first approximation and baseline of the dynamics and complexities entailed in the specific data environment of interest. Such first insights can often guide the researcher to choose additional NLP model classes that seem promising for the respective task. As suggested throughout this thesis, we then recommend following an NLP model agnostic research approach, in which the researcher ultimately evaluates a zoo of models on a dedicated validation set of their data. Unless a researcher has strong arguments to do otherwise, we suggest to let the data speak which model seems to be best suited for a research question at hand. Furthermore, evaluating over a model zoo allows the researcher to run model robustness tests and the researcher can also opt to average (also known as *ensembling*) over a set of models to avoid results being overly sensitive to individual model choices.

**Causal inference versus prediction:** For a successful research setup, it is also impor-

tant to initially clarify what the ultimate research goal is. In some cases, say in economic policy analysis, we are interested in discovering the true underlying causal effect - say the causal effect of an interest rate change on the real economy. In such case, we have to ensure that we use econometric and machine learning model setups that allow us to extract such a causal effect. Careful consideration of the research question, the choice of data, and the estimation methods is pivotal - particularly when we have to deal with observational data. Luckily, oftentimes prediction is all that is needed to answer our research question. Such realisation can potentially simplify a research project, since fewer constraints might be applying. As an example, in this thesis, we discussed two jointly supervised topic models - BTR and rSCHOLAR. Based on our results, BTR should be used over rSCHOLAR when causal inference of the regression coefficients is of importance. However, for pure prediction tasks the two models performed more similarly. One might want to opt for the computationally more efficient model (rSCHOLAR) in such a case, as causal inference capabilities are not needed.

**Incorporating domain knowledge into NLP model design:** In particular for modern large language models, the question presents itself how one can best incorporate domain specific economic and financial knowledge into models that have been trained on rather general datasets such as Wikipedia or the corpus of all English books. After all, language models can only be as good as the data they have been trained with. One way to integrate domain-specific knowledge into NLP models is to use transfer learning. Here, a model that was trained on general data is being further trained using domain-specific data. Domain specific models of BERT or GPT have been created for the financial domain - for example FinBERT (Araci, 2019), FlangBERT (Shah et al., 2022), or BloombergGPT (Wu et al., 2023). Further work on domain adaptations in economics and finance is certainly needed. For example, chapter 4 showed how even financial transformers struggled with multimodal macroeconomic and macrofinancial datasets. Further (potentially multimodal) pre-training and fine-tuning of large language models can enhance the *parametric knowledge* of a model. That is, the informational patterns from data that are stored in the parameter weights of a model. However, training and updating the parameters of large language models can be costly and time intensive. Especially, when the parameter size of a models ventures into the billions and when model updates might be required monthly,

weekly, or even daily. To alleviate this issue, one can also extend the knowledge accessible to a language model via external data sources. For instance, prompt based language models can be linked to *vector-stores*. In such a setup, instead of further pre-training or fine-tuning a language model on domain specific knowledge, one stores the the domain knowledge in a vectorised form (similar to word or document vectors described in this thesis) in a vector database. The language model has then access to search documents relevant to answer a specific prompt request in this vector-store. Information stored in such vector databases is also called *source knowledge*. These vector databases or vector-stores can be updated in much less cost and time intensive ways compared to an update of the entire large language model. This technique of allowing a language model to form its responses not only based on its parametric knowledge but also based on external source knowledge is called retrieval augmented generation (RAG) (Lewis et al., 2020). It is a very promising area of research to implement domain specific knowledge into modern NLP models. Future research in NLP for economic and financial modelling should very much consider RAG techniques in order to better leverage the power of large language models and domain specific knowledge.

**Practical considerations about NLP for economics and finance:** Finally, it is important to understand the practical limitations and potential biases when working with NLP models. **Data quality:** While powerful, NLP models are not infallible. The quality of input data is crucial. Incomplete or biased data can lead to misleading outcomes. **Continuous learning and adaptation:** The economic and financial landscapes are constantly evolving. Regular updates and adaptations of the models and their training data (and external source knowledge) are necessary to maintain their relevance and accuracy. **Ethical and responsible use:** Given the potential impact on markets and market participants, practitioners must use NLP tools ethically and responsibly, ensuring transparency and accountability in their methods. In conclusion, the application of NLP models in economics and finance offers substantial upside potential in terms of data analysis, prediction accuracy, and decision-making support. However, it requires a careful approach, considering the complexity of the data, the need for domain expertise, and the ethical implications of automated and 'black-box' decision-making.

# References


Maximilian Ahrens. Natural language processing and monetary policy. *MPhil Thesis, Oxford University*, 2018.

Maximilian Ahrens and Michael McMahon. Extracting Economic Signals from Central Bank Speeches. In *Proceedings of the Third Workshop on Economics and Natural Language Processing*, pages 93–114, 2021.

Maximilian Ahrens, Julian Ashwin, Jan-Peter Calliess, and Vu Nguyen. Bayesian Topic Regression for Causal Inference. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8162–8188, 2021. doi: 10.18653/v1/2021.emnlp-main.644.

Maximilian Ahrens, Deniz Erdemlioglu, Michael McMahon, Christopher J Neely, and Xiye Yang. Mind Your Language: Market Responses to Central Bank Speeches. *SSRN*, 2023.

Y. Aït-Sahalia and J. Jacod. Estimating the Degree of Activity of Jumps in High-Frequency Data. *The Annals of Statistics*, 37:2202–2244, 2009.

Jay Alammar. The Illustrated Transformer. *Blog Post*, 2018. URL https://jalammar.github.io/illustrated-transformer/.

T. G. Andersen, T. Bollerslev, F. X. Diebold, and H. Ebens. The Distribution of Realized Stock Return Volatility. *Journal of Financial Economics*, 61:43–76, 2001.

T. G. Andersen, T. Bollerslev, F. X. Diebold, and P Labys. Modeling and Forecasting Realized Volatility. *Econometrica*, 71:579–625, 2003.

T.G. Andersen, N. Fusari, and V. Todorov. The Pricing of Tail Risk and the Equity Premium: Evidence from International Option Markets. *Journal of Business and Economic Statistics*, 38:662–678, 2020.

Joshua D Angrist and Jörn-Steffen Pischke. *Mostly harmless econometrics: An empiricist's companion*. Princeton University Press, 2008.

Dolan Antenucci, Michael Cafarella, Margaret Levenstein, Christopher Ré, and Matthew D Shapiro. Using social media to measure labor market flows. Technical report, National Bureau of Economic Research, 2014.

Dogu Araci. FinBERT: Financial Sentiment Analysis with Pre-trained Language Models, 2019.

Boragan Aruoba and Thomas Drechsel. Identifying monetary policy shocks: A natural language approach. CEPR Discussion Papers 17133, C.E.P.R. Discussion Papers, March 2022.

Elliott Ash, Germain Gauthier, and Philine Widmer. Text Semantics Capture Political and Economic Narratives. *SSRN Electronic Journal*, 2021. doi: 10.2139/ssrn.3970603.

B. Coeuré. Policy analysis with big data, 2017. URL https://www.ecb.europa.eu/press/key/date/2017/html/ecb.sp171124.en.html.

BAAI. Wu Dao 2.0. *Blog Post*, 2018. URL https://gpt3demo.com/apps/wu-dao-20.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR*, 2015.

Scott R Baker, Nicholas Bloom, and Steven J Davis. Measuring economic policy uncertainty. *The quarterly journal of economics*, 131(4):1593–1636, 2016.

Tadas Baltrusaitis, Chaitanya Ahuja, and Louis Philippe Morency. Multimodal Machine Learning: A Survey and Taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):423–443, 2019. ISSN 19393539. doi: 10.1109/TPAMI.2018.2798607.

Abhijit V Banerjee. A simple model of herd behavior. *The quarterly journal of economics*, 107 (3):797–817, 1992.

S. M. Barakchian and C. Crowe. Monetary policy matters: Evidence from new shocks data. *Journal of Monetary Economics*, 60:950–966, 2013. doi: 10.1016/j.jmoneco.2013.09.006.

O. E. Barndorff-Nielsen and N. Shephard. Econometric Analysis of Realized Volatility and its Use in Estimating Stochastic Volatility Models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64:253–280, 2002.

Marvin J Barth III and Valerie A Ramey. The cost channel of monetary transmission. *NBER macroeconomics annual*, 16:199–240, 2001.

M. D. Bauer, A. Lakdawala, and P. Mueller. Market-Based Monetary Policy Uncertainty. *The Economic Journal*, 132:1290–1308, 2022.

G. Bekaert, M. Hoerova, and M. L. Duca. Risk, Uncertainty and Monetary Policy. *Journal of Monetary Economics*, 60:771–788, 2013.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The Long-Document Transformer. *arXiv*, 2020.

Ben S Bernanke. Alternative Explanations of the Money-Income Correlation. *NBER Working Paper Series*, (1842), 1986.

Ben S Bernanke and Kenneth N Kuttner. What Explains the Stock Market's Reaction to Federal Reserve Policy? *Journal of Finance*, 60(3):1221–1257, 2005.

Ben S. Bernanke, Jean Boivin, and Piotr S Eliasz. Measuring the Effects of Monetary Policy: A Factor-augmented Vector Autoregressive (FAVAR) Approach. *The Quarterly Journal of Economics*, 120(1):387–422, 1 2005.

David Bholat, Stephen Hansen, Pedro Santos, and Cheryl Schonhardt-Bailey. Text mining for central banks. *Centre for Central Banking Studies*, (33), 2015.

Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

David Blei. Probabilistic topic models. *Communications of the ACM*, 55:77–84, 2012. ISSN 00010782. doi: 10.1145/2133806.2133826.

David Blei and John Lafferty. Dynamic Topic Models. In *Proceedings of the 23rd Internaional Conference on Machine Learning*, pages 113–120, 2006.

David M Blei and John D Lafferty. Correlated topic models. In *Advances in Neural Information Processing Systems*, pages 147–154, 2005. ISBN 9780262232531.

David M Blei and Jon D McAuliffe. Supervised topic models. *Advances in neural information processing systems*, pages 121–128, 2008.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

David M Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational Inference: A Review for Statisticians, 2017. ISSN 1537274X.

David M Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational Inference: A Review for Statisticians, 2018. ISSN 1537274X.

Alan S Blinder. Through a Crystal Ball Darkly: The Future of Monetary Policy Communication. *AEA Papers and Proceedings*, 108:567–71, 2018. ISSN 2574-0768. doi: 10.1257/pandp.20181080.

Alan S Blinder, Michael Ehrmann, Marcel Fratzscher, Jakob De Haan, and David Jan Jansen. Central bank communication and monetary policy: A survey of theory and evidence. *Journal of Economic Literature*, 46(4):910–945, 2008. ISSN 00220515. doi: 10.1257/jel.46.4.910.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5: 135–146, 2017. doi: 10.1162/tacl{_}a{_}00051.

T. Bollerslev, J. Li, and Y. Xue. Volume, Volatility, and Public Announcements. *The Review of Economic Studies*, 85:2005–2041, 2018.

H. P. Boswijk, R. J. A. Laeven, and X. Yang. Testing for Self-Excitation in Jumps. *Journal of Econometrics*, 203:256–266, 2018.

Mark Boukes, Bob van de Velde, Theo Araujo, and Rens Vliegenthart. What's the Tone? Easy Doesn't Do It: Analyzing Performance and Agreement Between Off-the-Shelf Sentiment Analysis Tools. *Communication Methods and Measures*, 14(2):83–104, 2020. ISSN 19312466. doi: 10.1080/19312458.2019.1671966.

Ellyn Boukus and Joshua Rosenberg. The information content of FOMC minutes. Technical report, Federal Reserve Bank of New York, 2006.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz

Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam Mccandlish, Alec Radford, Ilya Sutskever, and Dario Amodei Openai. Language Models are Few-Shot Learners. 2020.

David Byrne, Robert Goodhead, Michael McMahon, and Conor Parle. The central bank crystal ball: Temporal information in monetary policy communication. Discussion Papers 17930, Centre for Economic Policy Research, February 2023.

Ricardo J. Caballero and Alp Simsek. Monetary policy with opinionated markets. *American Economic Review*, 112(7):2353–92, July 2022.

Yong Cai, Santiago Camara, and Nicholas Capel. It's not always about the money, sometimes it's about sending a message: Evidence of Informational Content in Monetary Policy Announcements. pages 1–24, 2021.

Kris Cao and Laura Rimell. You should evaluate your language model on marginal likelihood over tokenisations. In *EMNLP 2021 - 2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, pages 2104–2114, 2021. ISBN 9781955917094. doi: 10.18653/v1/2021.emnlp-main.161.

Dallas Card, Chenhao Tan, and Noah A. Smith. Neural models for documents with metadata. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2031–2040. Association for Computational Linguistics, July 2018.

Carlos Carvalho, Eric Hsu, and Fernanda Nechio. Measuring the effect of the zero lower bound on monetary policy. Technical Report 2016-6, Federal Reserve Bank of San Francisco, 4 2016.

Jianshu Chen, Ji He, Yelong Shen, Lin Xiao, Xiaodong He, Jianfeng Gao, Xinying Song, and Li Deng. End-to-end learning of lda by mirror-descent back propagation over a deep architecture. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28, 2015.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating Long Sequences with Sparse Transformers. Technical report, 2019. URL https://openai.com/blog/sparse-transformer.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling Language Modeling with Pathways. pages 1–87, 2022.

Lawrence J Christiano, Martin Eichenbaum, and Charles L Evans. Monetary policy shocks: What have we learned and to what end? In J B Taylor and M Woodford, editors, *Handbook of Macroeconomics*, volume 1 of *Handbook of Macroeconomics*, chapter 2, pages 65–148. Elsevier, 1999.

Lawrence J. Christiano, Martin Eichenbaum, and Charles L. Evans. Nominal rigidities and the dynamic effects of a shock to monetary policy. *Journal of Political Economy*, 113(1):1–45, 2005. ISSN 00223808. doi: 10.1086/426038.

A. Cieslak and A. Schrimpf. Non-Monetary News in Central Bank Communication. *Journal of International Economics*, 118:293–315, 2019.

Anna Cieslak and Michael McMahon. Tough talk: The Fed and the risk premia. Mimeograph, 2023.

Anna Cieslak, Stephen Hansen, Michael McMahon, and Song Xiao. Policymakers' uncertainty. Mimeograph, 2023.

Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. CANINE: Pre-training an Efficient Tokenization-Free Encoder for Language Representation. *Transactions of the Association for Computational Linguistics*, 10:73–91, 2022. ISSN 2307387X. doi: 10.1162/tacl{_}a{_}00448.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D Manning. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. 2020.

John H. Cochrane and Monika Piazzesi. The fed and interest rates - a high-frequency identification. *American Economic Review*, 92(2):90–95, May 2002. doi: 10.1257/000282802320189069.

Olivier Coibion and Yuriy Gorodnichenko. Information rigidity and the expectations formation process: A simple framework and new facts. *American Economic Review*, 105(8):2644–2678, 2015.

Jurafsky Daniel and James H Martin. Speech and Language Processing Chapter Three: N-gram Language Models. 2021.

Sanjiv Das, Connor Goggins, John He, George Karypis, Sandeep Krishnamurthy, Mitali Mahajan, Nagpurnanand Prabhala, Dylan Slack, Rob Van Dusen, Shenghua Yue, Sheng Zha, and Shuai Zheng. Context, Language Modeling, and Multimodal Data in Finance.

Sanjiv R. Das, Michele Donini, Muhammad Bilal Zafar, John He, and Krishnaram Kenthapadi. FinLex: An effective use of word embeddings for financial lexicon generation. *Journal of Finance and Data Science*, 8:1–11, 2022. ISSN 24059188. doi: 10.1016/j.jfds.2021.10.001.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data Via the EM Algorithm . *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977. doi: 10.1111/j.2517-6161.1977.tb01600.x.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2018.

Mario Draghi. Remarks at central bank communications conference, 2017. URL https://www.youtube.com/watch?v=DI7p-g5108g.

Sahibsingh A Dudani. The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, (4):325–327, 1976.

M. Dungey, D. Erdemlioglu, M. Matei, and X. Yang. Testing for Mutually Exciting Jumps and Financial Flights in High Frequency Data. *Journal of Econometrics*, 202:18–44, 2018.

Naoki Egami, Christian J Fong, Justin Grimmer, Margaret E Roberts, and Brandon M Stewart. How to make causal inferences using texts. *arXiv preprint arXiv:1802.02163*, 2018.

M. Ehrmann and J. Talmi. Starting from a Blank Page? Semantic Similarity in Central Bank Communication and Market Volatility. *Journal of Monetary Economics*, 111:48–62, 2020.

Jacob Eisenstein, Amr Ahmed, and Eric P Xing. Sparse additive generative models of text. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 1041–1048. Citeseer, 2011.

Saskia Ter Ellen, Vegard H. Larsen, and Leif Anders Thorsrud. Narrative monetary policy surprises and the media. *Journal of Money, Credit and Banking*, 54(5):1525–1549, 2022.

D. Erdemlioglu and X. Yang. News Arrival, Time-Varying Jump Intensity, and Realized Volatility: Conditional Testing Approach. *Journal of Financial Econometrics*, (nbac015): 1–38, 2022.

Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data. 2020.

Fed. Nontraditional Data, Machine Learning, and Natural Language Processing in Macroeconomics, 2019.

William Fedus, Barret Zoph, and Noam Shazeer. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research*, 23:1–40, 2022. ISSN 15337928.

John R Firth. A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, 1957.

Christian Fong and Justin Grimmer. Discovery of treatments from text corpora. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1600–1609, Berlin, Germany, August 2016. Association for Computational Linguistics.

Ragnar Frisch and Frederick V Waugh. Partial time regressions as compared with individual trends. *Econometrica: Journal of the Econometric Society*, pages 387–401, 1933.

Matthew Gentzkow, Bryan Kelly, and Matt Taddy. Text as Data. *Journal of Economic Literature*, 57(3):535–574, 2019.

Mark Gertler and Peter Karadi. Monetary policy surprises, credit costs, and economic activity. *American Economic Journal: Macroeconomics*, 7(1):44–76, 2015. ISSN 19457715. doi: 10.1257/mac.20130329.

Pavel Gertler and Roman Horvath. Central bank communication and financial markets: New high-frequency evidence. *Journal of Financial Stability*, 36(C):336–345, 2018.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.

R. Gómez-Cram and M. Grotteria. Real-Time Price Discovery via Verbal Communication: Method and Application to Fedspeak. *Journal of Financial Economics*, 143:993–1025, 2022.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Yuriy Gorodnichenko and Michael Weber. Are Sticky Prices Costly? Evidence from the Stock Market †. *American Economic Review*, 106(1):165–199, 2016. doi: 10.1257/aer.20131513.

Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235, 2004.

Justin Grimmer and Brandon M Stewart. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political analysis*, 21(3):267–297, 2013.

Refet S Gürkaynak, Brian Sack, and Eric Swanson. Do Actions Speak Louder Than Words? The Response of Asset Prices to Monetary Policy Actions and Statements. *International Journal of Central Banking*, 1(1), 5 2005.

Refet S Gürkaynak, Brian Sack, and Eric Swanson. The sensitivity of long-term interest rates to economic news: Evidence and implications for macroeconomic models. *American economic review*, 95(1):425–436, 2005.

Refet S Gurkaynak, Brian P Sack, and Eric T Swanson. Do actions speak louder than words? the response of asset prices to monetary policy actions and statements. *International Journal of Central Banking*, 1(1), 2005.

Andy Haldane and Michael McMahon. Central Bank Communication and the General Public. *AEA Papers and Proceedings*, 1(1):Forthcoming, 2018. ISSN 2574-0768.

Amy Handlan. Text shocks and monetary surprises: Text analysis of fomc statements with machine learning. *Published Manuscript*, 2020.

S. Hansen, M. McMahon, and A. Prat. Transparency and Deliberation within the FOMC: A Computational Linguistics Approach. *The Quarterly Journal of Economics*, 133:801–870, 2018.

Stephen Hansen and Michael McMahon. Shocking language: Understanding the macroeconomic effects of central bank communication. *Journal of International Economics*, 99:114–133, 2016.

Stephen Hansen, Michael McMahon, and Matthew Tong. The long-run information effect of central bank communication. *Journal of Monetary Economics*, 108:185–202, 2019. ISSN 03043932. doi: 10.1016/j.jmoneco.2019.09.002.

Samuel G. Hanson and Jeremy C. Stein. Monetary policy and long-term real rates. *Journal of Financial Economics*, 115(3):429–448, 2015. ISSN 0304405X. doi: 10.1016/j.jfineco.2014.11.001.

M. Hattori, A. Schrimpf, and V. Sushko. The Response of Tail Risk Perceptions to Unconventional Monetary Policy. *American Economic Journal: Macroeconomics*, 8:111–136, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016. ISSN 03764699. doi: 10.1002/chin.200650130.

B. M. Hill. A Simple General Approach to Inference about The Tail of a Distribution. *The Annals of Statistics*, 3:1163–1174, 1975.

G.E. Hinton, J. L. McCleland, and RumelhartD.E. Distributed representations. *Parallel distributed processing: Explorations in the microstructure of cognition. Foundations. MIT Press*, 1, 1986.

Matthew D Hoffman, David M Blei, Chong Wang, John Paisley, Jpaisley@berkeley Edu, and Tommi Jaakkola. Stochastic Variational Inference. In *Journal of Machine Learning Research*, volume 14, pages 1303–1347, 2013.

Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 289–296. Morgan Kaufmann Publishers Inc., 1999.

L. Husted, J. Rogers, and B. Sun. Monetary Policy Uncertainty. *Journal of Monetary Economics*, 115:20–36, 2020.

Marek Jarociński and Peter Karadi. Deconstructing monetary policy surprises-The role of information shocks. *American Economic Journal: Macroeconomics*, 12(2):1–43, 2020. ISSN 19457715. doi: 10.1257/mac.20180090.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

Katherine Keith, David Jensen, and Brendan O'Connor. Text and causal inference: A review of using text to remove confounding from causal estimates. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5332–5344, Online, July 2020. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR, Conference Track Proceedings*, 2015.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR, Conference Track Proceedings*, 2014.

Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, volume 1, pages 66–75, 2018. ISBN 9781948087322. doi: 10.18653/v1/p18-1007.

Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *EMNLP 2018 - Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Proceedings*, pages 66–71, 2018. ISBN 9781948087858. doi: 10.18653/v1/d18-2012.

Kenneth N Kuttner. Monetary policy surprises and interest rates: Evidence from the Fed funds futures market. *Journal of monetary economics*, 47(3):523–544, 2001.

Vegard H. Larsen and Leif A. Thorsrud. The value of news for economic developments. *Journal of Econometrics*, 2019. ISSN 18726895. doi: 10.1016/j.jeconom.2018.11.013.

Markus Leippold. Sentiment Spin : Attacking Financial Sentiment with. pages 1–24, 2023.

M. Leombroni, A. Vedolin, G. Venter, and P. Whelan. Central Bank Communication and the Yield Curve. *Journal of Financial Economics*, 141:860–880, 2021.

Richard A Levine and George Casella. Implementations of the monte carlo em algorithm. *Journal of Computational and Graphical Statistics*, 10(3):422–439, 2001.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

Kai Li, Feng Mai, Rui Shen, and Xinyan Yan. Measuring Corporate Culture Using Machine Learning. *Review of Financial Studies*, 34(7):3265–3315, 2021. ISSN 14657368. doi: 10.1093/rfs/hhaa079.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov, and Paul G Allen. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*, 2019.

Tim Loughran and Bill Mcdonald. When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks. *Journal of Finance*, 66(1):35–65, 2011. ISSN 00221082. doi: 10.1111/j.1540-6261.2010.01625.x.

Tim Loughran and Bill McDonald. When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *The Journal of Finance*, 66(1):35–65, 2011.

Michael C Lovell. Seasonal adjustment of economic time series and multiple regression analysis. *Journal of the American Statistical Association*, 58(304):993–1010, 1963.

Michael C Lovell. A simple proof of the fwl theorem. *The Journal of Economic Education*, 39 (1):88–91, 2008.

David O Lucca and Francesco Trebbi. Measuring Central Bank Communication: An Automated Approach with Application to FOMC Statements. *NBER Working Paper Series*, (15367), 9 2009.

Måns Magnusson, Leif Jonsson, and Mattias Villani. Dolda: a regularized supervised topic model for high-dimensional multi-class regression. *Computational Statistics*, 35(1):175–201, 2020.

Pekka Malo, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65(4):782–796, 2014. ISSN 23301643. doi: 10.1002/asi.23062.

Martin Mandler. Inflation-regime dependent effects of monetary policy shocks. Evidence from threshold vector autoregressions. *Economics Letters*, 116(3):422–425, 2012. ISSN 01651765. doi: 10.1016/j.econlet.2012.04.027.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. 2008. ISBN 9780521865715. doi: 10.1002/asi.21234.

Deirdre Nansen McCloskey. Adam Smith did Humanomics: So Should We. *Eastern Economic Journal*, 42(4):503–513, 2016.

Donald McCloskey and Arjo Klamer. One Quarter of GDP Is Persuasion. *Source: The American Economic Review*, 85(2):191–195, 1995.

Yishu Miao, Lei Yu, and Phil Blunsom. Neural variational inference for text processing. In *33rd International Conference on Machine Learning, ICML 2016*, volume 4, pages 2589–2600, 2016. ISBN 9781510829008.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. Discovering discrete latent topics with neural variational inference. In *34th International Conference on Machine Learning, ICML 2017*, volume 5, pages 3721–3731, 2017. ISBN 9781510855144.

Sabrina J Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y Lee, Benoît Sagot, and Samson Tan. Between words and characters: A Brief History of Open-Vocabulary Modeling and Tokenization in NLP. 2021.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, 2013a.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic Regularities in Continuous Space Word Representations. *NAACL*, pages 746–751, 2013b.

Silvia Miranda-Agrippino and Giovanni Ricco. The transmission of monetary policy shocks. *American Economic Journal: Macroeconomics*, 13(3):74–107, July 2021.

Reagan Mozer, Luke Miratrix, Aaron Russell Kaufman, and L. Jason Anastasopoulos. Matching with text data: An experimental evaluation of methods for matching documents and of measuring match quality. *Political Analysis*, 28(4):445–468, 2020. ISSN 14764989. doi: 10.1017/pan.2020.1.

Hannes Mueller and Christopher Rauh. Reading between the lines: Prediction of political violence using newspaper text. *American Political Science Review*, 112(2):358–375, 2018.

Emi Nakamura and Jón Steinsson. High-frequency identification of monetary non-neutrality: The information effect. *Quarterly Journal of Economics*, 133(3):1283–1330, 2018. ISSN 15314650. doi: 10.1093/QJE/QJY004.

James Nesbit. Text as Instruments * (Job Market Paper) Click here for latest version. Technical report, 2020.

Andreas Neuhierl and Michael Weber. Monetary policy communication, policy slope, and the stock market. *Journal of Monetary Economics*, 108:140–155, 2019.

A. C.R. Ochs. A New Monetary Policy Shock with Text Analysis. Cambridge working papers in economics, Faculty of Economics, University of Cambridge, June 2021.

OpenAI. chatGPT. *Website*, 2022. URL https://openai.com/blog/chatgpt.

OpenAI. GPT-4 Technical Report. *Technical Report*, 2023.

A. Ozdagli and M. Velikov. Show Me the Money: The Monetary Policy Risk Premium. *Journal of Financial Economics*, 135:320–339, 2020.

Judea Pearl. *Causality (2nd edition)*. Cambridge University Press, 2009.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Musicassette Interchangeability. the Facts Behind the Facts. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014.

Anastasios Petropoulos and Vasilis Siakoulis. Can central bank speeches predict financial market turbulence? Evidence from an adaptive NLP sentiment index analysis using XGBoost machine learning technique. *Central Bank Review*, 21(4):141–153, 2021. ISSN 25241699. doi: 10.1016/j.cbrev.2021.12.002.

Mary Phuong and Marcus Hutter. Formal Algorithms for Transformers. *arXiv*, (July):1–16, 2022.

Monika Piazzesi and Eric T Swanson. Futures prices as risk-adjusted forecasts of monetary policy. *Journal of Monetary Economics*, 55(4):677–691, 2008.

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

Danish Pruthi, Bhuwan Dhingra, and Zachary C Lipton. Combating adversarial misspellings with robust word recognition. In *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, pages 5582–5591, 2019. ISBN 9781950737482. doi: 10.18653/v1/p19-1561.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. *OpenAI Blog 1, no. 8*, 2019. URL https://github.com/codelucas/newspaper.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21:1–67, 2020.

V. A. Ramey. Macroeconomic Shocks and Their Propagation. In *Handbook of Macroeconomics*, volume 2, pages 71–162. 2016. ISBN 9780444594877. doi: 10.1016/bs.hesmac.2016.03.003.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR.

Christian Robert and George Casella. *Monte Carlo statistical methods*. Springer Science & Business Media, 2013.

Margaret E Roberts, Brandon M Stewart, Dustin Tingley, Christopher Lucas, Jetson Leder-Luis, Shana Kushner Gadarian, Bethany Albertson, and David G Rand. Structural topic models for open-ended survey responses. *American Journal of Political Science*, 58(4):1064–1082, 2014.

Margaret E Roberts, Brandon M Stewart, and Edoardo M Airoldi. A model of text for experimentation in the social sciences. *Journal of the American Statistical Association*, 111 (515):988–1003, 2016.

Margaret E Roberts, Brandon M Stewart, and Richard A Nielsen. Adjusting for confounding with text matching. *American Journal of Political Science*, 64(4):887–903, 2020.

Christina Romer and David H Romer. A New Measure of Monetary Shocks: Derivation and Implications. *American Economic Review*, 94(4):1055–1084, 2004. ISSN 0002-8282. doi: 10.1257/0002828042002651.

Christina D. Romer and David H. Romer. Does Monetary Policy Matter? A New Test in the Spirit of Friedman and Schwartz. *NBER Macroeconomics Annual*, 4:121–170, 1 1989. ISSN 0889-3365. doi: 10.1086/654103.

Christina D Romer and David H Romer. Federal Reserve Information and the Behavior of Interest Rates. *American Economic Review*, 90(3):429–457, 6 2000.

Sasha Rush, Austin Huang, Suraj Subramanian, Jonathan Sum, Khalid Almubarak, and Biderman Stella. Annotated Transformer. *Blog Post*, 2018. URL http://nlp.seas.harvard.edu/annotated-transformer/#conclusion.

Cheryl Schonhardt-Bailey. *Deliberating Monetary Policy*. MIT Press, Cambridge, 2013.

Mike Schuster and Nakajima Kaisuke. Japanese and Korean Voice Search. In *IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 2012. ISBN 9781955917094. doi: 10.18653/v1/2021.emnlp-main.160.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers*, volume 3, pages 1715–1725, 2016. ISBN 9781510827585. doi: 10.18653/v1/p16-1162.

Raj Sanjay Shah, Kunal Chawla, Dheeraj Eidnani, Agam Shah, Wendi Du, Sudheer Chava, Natraj Raman, Charese Smiley, Jiaao Chen, and Diyi Yang. WHEN FLUE MEETS FLANG: Benchmarks and Large Pre-trained Language Model for Financial Domain. 2022.

Adam Hale Shapiro, Moritz Sudhof, and Daniel J Wilson. Measuring news sentiment. *Journal of econometrics*, 228(2):221–243, 2022.

Robert J Shiller. Narrative Economics. *American Economic Review*, 107(44), 2017. doi: 10.1257/aer.107.4.967.

Takahiro Shinozaki and Mari Ostendorf. Cross-validation em training for robust parameter estimation. In *IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP*, volume 4, pages IV–437, 2007.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism. 9 2019.

Frank Smets and Rafael Wouters. Shocks and Frictions in US Business Cycles: A Bayesian DSGE Approach. *American Economic Review*, 97(3):586–606, 2007. ISSN 00028282. doi: 10.1257/aer.97.3.586.

Akash Srivastava and Charles Sutton. Neural Variational Inference For Topic Models. *NIPS 2016, Workshop on Bayesian Deep Learning*, 2016.

Akash Srivastava and Charles Sutton. Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*, 2017.

Joe Staines. Mining Text and Time Series Data with Applications in Finance. Technical report, 2014.

Mark Steyvers and Tom Griffiths. Probabilistic Topic Models. In *Latent Semantic Analysis: A Road To Meaning*. 2007. ISBN 1532-4435. doi: 10.1016/s0364-0213(01)00040-4.

Philip J Stone, Dexter C Dunphy, and Marshall S Smith. The general inquirer: A computer approach to content analysis. 1966.

Alan Stuart, Steven Arnold, J Keith Ord, Anthony O'Hagan, and Jonathan Forster. *Kendall's advanced theory of statistics*. Wiley, 1994.

Lichao Sun, Kazuma Hashimoto, Wenpeng Yin, Akari Asai, Jia Li, Philip Yu, and Caiming Xiong. Adv-BERT: BERT is not robust on misspellings! Generating nature adversarial samples on BERT. *arXiv*, 2020.

E. T. Swanson. Measuring the effects of federal reserve forward guidance and asset purchases on financial markets. *Journal of Monetary Economics*, 118:32–53, 2021.

Eric T. Swanson. The importance of fed chair speeches as a monetary policy tool. *AEA Papers and Proceedings*, 113:394–400, May 2023.

Chenhao Tan, Lillian Lee, and Bo Pang. The effect of wording on message propagation: Topic- and author-controlled natural experiments on Twitter. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 175–185, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

Yi Tay, Vinh Q. Tran, Sebastian Ruder, Jai Gupta, Hyung Won Chung, Dara Bahri, Zhen Qin, Simon Baumgartner, Cong Yu, and Donald Metzler. Charformer: Fast Character Transformers via Gradient-based Subword Tokenization. pages 1–20, 2021.

Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. Sharing clusters among related groups: Hierarchical dirichlet processes. *Advances in Neural Information Processing Systems*, (1), 2005. ISSN 10495258.

Silvana Tenreyro and Gregory Thwaites. Pushing on a string: US monetary policy is less powerful in recessions. *American Economic Journal: Macroeconomics*, 8(4):43–74, 2016.

P. Tillmann. Monetary Policy Uncertainty and the Response of the Yield Curve to Policy Shocks. *Journal of Money, Credit and Banking*, 54:803–833, 2020.

Wouter van Atteveldt, Mariken A.C.G. van der Velden, and Mark Boukes. The Validity of Sentiment Analysis: Comparing Manual Annotation, Crowd-Coding, Dictionary Approaches, and Machine Learning Algorithms. *Communication Methods and Measures*, 15(2):121–140, 2021. ISSN 19312466. doi: 10.1080/19312458.2020.1869198.

C J Van Rijsbergen, S E Robertson, and M F Porter. New models in probabilistic information retrieval. *British Library Research and Development Report*, 5587(5587):123, 1980.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

Victor Veitch, Dhanya Sridhar, and David Blei. Adapting text embeddings for causal inference. In *Conference on Uncertainty in Artificial Intelligence*, pages 919–928. PMLR, 2020.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *Advances in Neural Information Processing Systems*, 2019a. ISSN 10495258.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proceedings of ICLR*, 2019b. doi: 10.1097/00006199-198905000-00001.

Xinyi Wang and Yi Yang. Neural topic model with attention for supervised learning. In *International Conference on Artificial Intelligence and Statistics*, pages 1147–1156. PMLR, 2020.

Yining Wang and Jun Zhu. Spectral methods for supervised topic models. In *Advances in Neural Information Processing Systems*, pages 1511–1519, 2014.

Nicolas Webersinke, Mathias Kraus, Julia Bingler, and Markus Leippold. ClimateBERT: A Pretrained Language Model for Climate-Related Text. In *Proceedings of AAAI 2022 Fall Symposium: The Role of AI in Responding to Climate Challenges*, 2022. doi: https://doi.org/10.48550/arXiv.2212.13631.

Zach Wood-Doughty, Ilya Shpitser, and Mark Dredze. Challenges of using text classifiers for causal inference. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, volume 2018, page 4586. NIH Public Access, 2018.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*, 2023.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv*, 2016.

Frank Z Xing, Erik Cambria, and Roy E Welsch. Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, 50(1):49–73, 2018.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. ByT5: Towards a Token-Free Future with Pre-trained Byte-to-Byte Models. *Transactions of the Association for Computational Linguistics*, 10:291–306, 2022. ISSN 2307387X. doi: 10.1162/tacl{_}a{_}00461.

Janet Yellen. Remarks at Central Bank Communications Conference, ECB, 14 November 2017. \url{https://www.youtube.com/watch?v=DI7p-g51O8g}, 2017.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems*, volume 2020-Decem, 2020.

Jun Zhu, Amr Ahmed, and Eric P Xing. Medlda: maximum margin supervised topic models. *Journal of Machine Learning Research*, 13(Aug):2237–2278, 2012.

# Appendices

# A

# Appendix: EcoFinBench - A Natural Language Processing Benchmark for Economics and Finance

## A.1 AutoGluon - Machine Learning Model Zoo

AutoGluon is an automated machine learning (AutoML) framework that has been developed to fuse multimodal features such as text, images, and tabular data. We chose this AutoML framework because it outperformed competing frameworks in multimodal benchmark tasks (see Erickson et al. (2020)).

**Base models**: AutoGluon fits machine learning *base models* and then combines them through ensembling and stacking to boost performance. AutoGluon allows us to apply hyperparameter optimization over all models. The *base models* in AutoGluon span the following broad machine learning algorithm classes:

1. **K-nearest neighbours** (Dudani, 1976): AutoGluon uses two variations of k-nearest neighbours (KNN) that differ in their weighting approaches. One allocates uniform weights to all points while the other weights points according to the inverse of their respective distances.

2. **Random forests** (Breiman, 2001): AutoGluon again deploys two variations of this algorithm class. One option uses the information gain of nodes for the assessment of the split quality. The other option uses Gini impurity instead.

3. **Extremely randomized trees** (Geurts et al., 2006): For the random tree class, AutoGluon deploys both an implementation resorting to information gain and another option that uses Gini impurity for the assessment of split quality.

4. **Boosted decision trees**: AutoGluon runs (where applicable to the task) Extreme Gradient Boosting (Chen and Guestrin, 2016), Light Gradient Boosting (Ke et al., 2017), Categorical Boosting (Prokhorenkova et al., 2018).

5. **Neural networks**: A detailed description of the the neural network architecture can be found in **?**. The architecture has been specifically designed for the multimodal use of categorical (text, images) and numerical data. It uses variable-specific embeddings for each of the categorical features. These are then concatenated with the numerical features into one overall input vector. This vector is in turn fed through a 3-layer feed-forward network as well as through a linear skip-connection. Model ensembling and stacking can be applied and are optimally chosen in the validation process.

## A.2 Text-only Datasets: Detailed Results

### A.2.1 Bluebooks

Table A.1: FOMC Bluebooks Alternatives benchmark

| model | macro F1 | model | macro F1 |
|---|---|---|---|
| Tabpred:dtm_XGBoost_BAG_L1 | 96.4 (2.8) | LDA-K500 | 76.8 (7.8) |
| FinBERT | 96.0 (5.0) | Tabpred:lm_XGBoost_BAG_L1 | 76.4 (7.1) |
| Tabpred:dtm_LightGBMLarge_BAG_L1 | 95.9 (3.1) | LDA-K250 | 75.9 (7.4) |
| Tabpred:dtm_CatBoost_BAG_L1 | 95.7 (2.9) | Tabpred:lm_RandomForestEntr_BAG_L1 | 75.9 (8.3) |
| Tabpred:dtm_WeightedEnsemble_L2 | 95.7 (3.7) | Tabpred:lm_WeightedEnsemble_L2 | 75.4 (6.3) |
| Tabpred:dtm_LightGBM_BAG_L1 | 95.1 (3.7) | Tabpred:lm_ExtraTreesEntr_BAG_L1 | 74.5 (7.7) |
| Tabpred:dtm_LightGBMXT_BAG_L1 | 95.0 (3.4) | Tabpred:lm_ExtraTreesGini_BAG_L1 | 74.2 (8.4) |
| Tabpred:dtm_NeuralNetTorch_BAG_L1 | 94.2 (4.3) | Tabpred:lm_LightGBMLarge_BAG_L1 | 73.9 (6.8) |
| WordCount-lin | 92.4 (13.8) | Tabpred:lm_LightGBM_BAG_L1 | 73.7 (6.3) |
| Tabpred:dtm_RandomForestEntr_BAG_L1 | 92.1 (4.3) | Tabpred:lm_LightGBMXT_BAG_L1 | 71.1 (8.1) |
| Tabpred:dtm_ExtraTreesEntr_BAG_L1 | 92.1 (4.3) | LDA-K100 | 70.5 (8.1) |
| Tabpred:dtm_ExtraTreesGini_BAG_L1 | 92.0 (4.2) | Tabpred:lm_NeuralNetTorch_BAG_L1 | 70.4 (8.4) |
| Tabpred:dtm_RandomForestGini_BAG_L1 | 91.2 (5.0) | Tabpred:lm_NeuralNetFastAI_BAG_L1 | 69.5 (8.5) |
| FLANG-BERT | 91.0 (9.2) | LDA-K50 | 67.8 (8.9) |
| BERT-base | 86.6 (12.6) | LMdict-lin | 64.6 (6.6) |
| Tabpred:dtm_NeuralNetFastAI_BAG_L1 | 84.3 (5.5) | LMdict-naive | 63.3 (6.5) |
| Tabpred:dtm_KNeighborsDist_BAG_L1 | 81.1 (5.1) | Tabpred:lm_KNeighborsDist_BAG_L1 | 56.5 (7.0) |
| Tabpred:dtm_KNeighborsUnif_BAG_L1 | 79.7 (5.4) | Tabpred:lm_KNeighborsUnif_BAG_L1 | 54.4 (5.8) |
| LDA-K1000 | 77.7 (8.2) | LDA-K10 | 50.2 (10.6) |
| Tabpred:lm_RandomForestGini_BAG_L1 | 77.1 (7.7) | LDA-K5 | 40.1 (7.8) |
| Tabpred:lm_CatBoost_BAG_L1 | 77.0 (7.1) | majority class | 28.8 (0.8) |

Avg. over 50 runs, standard deviation in brackets, train size: 327, test size: 82 (shuffled)

### A.2.2 Twitter Financial News

Table A.2: Twitter Financial News benchmark

| model | macro F1 | model | macro F1 |
|---|---|---|---|
| FLANG-BERT | 82.2 (2.9) | Tabpred:lm_ExtraTreesGini_BAG_L1 | 45.9 (0.0) |
| FinBERT | 81.5 (1.3) | Tabpred:lm_LightGBM_BAG_L2 | 45.9 (0.0) |
| BERT-base | 80.8 (1.7) | Tabpred:lm_RandomForestGini_BAG_L1 | 45.9 (0.0) |
| Tabpred:dtm_LightGBMLarge_BAG_L2 | 77.7 (0.0) | Tabpred:lm_ExtraTreesEntr_BAG_L1 | 45.7 (0.0) |
| Tabpred:dtm_XGBoost_BAG_L2 | 77.2 (0.0) | Tabpred:lm_RandomForestEntr_BAG_L1 | 45.6 (0.0) |
| Tabpred:dtm_NeuralNetFastAI_BAG_L2 | 76.9 (0.0) | Tabpred:lm_LightGBMXT_BAG_L2 | 45.6 (0.0) |
| Tabpred:dtm_RandomForestGini_BAG_L2 | 76.8 (0.0) | Tabpred:lm_WeightedEnsemble_L3 | 45.6 (0.0) |
| Tabpred:dtm_CatBoost_BAG_L2 | 76.7 (0.0) | Tabpred:lm_LightGBMLarge_BAG_L1 | 45.5 (0.0) |
| Tabpred:dtm_ExtraTreesEntr_BAG_L2 | 76.7 (0.0) | Tabpred:lm_LightGBMLarge_BAG_L2 | 45.0 (0.0) |
| Tabpred:dtm_LightGBM_BAG_L2 | 76.6 (0.0) | Tabpred:lm_ExtraTreesGini_BAG_L2 | 45.0 (0.0) |
| Tabpred:dtm_WeightedEnsemble_L3 | 76.6 (0.0) | Tabpred:lm_RandomForestGini_BAG_L2 | 44.9 (0.0) |
| Tabpred:dtm_LightGBMXT_BAG_L2 | 76.6 (0.0) | Tabpred:lm_LightGBM_BAG_L1 | 44.9 (0.0) |
| Tabpred:dtm_RandomForestEntr_BAG_L2 | 76.5 (0.0) | Tabpred:lm_XGBoost_BAG_L1 | 44.8 (0.0) |
| Tabpred:dtm_ExtraTreesGini_BAG_L2 | 76.0 (0.0) | Tabpred:lm_RandomForestEntr_BAG_L2 | 44.7 (0.0) |
| Tabpred:dtm_WeightedEnsemble_L2 | 75.8 (0.0) | Tabpred:lm_ExtraTreesEntr_BAG_L2 | 44.4 (0.0) |
| Tabpred:dtm_LightGBMLarge_BAG_L1 | 74.1 (0.0) | Tabpred:lm_LightGBMXT_BAG_L1 | 44.4 (0.0) |
| Tabpred:dtm_XGBoost_BAG_L1 | 72.3 (0.0) | Tabpred:dtm_KNeighborsDist_BAG_L1 | 44.1 (0.0) |
| Tabpred:dtm_CatBoost_BAG_L1 | 69.5 (0.0) | Tabpred:lm_CatBoost_BAG_L1 | 43.3 (0.0) |
| Tabpred:dtm_LightGBM_BAG_L1 | 69.1 (0.0) | Tabpred:lm_NeuralNetTorch_BAG_L1 | 42.5 (0.0) |
| Tabpred:dtm_RandomForestGini_BAG_L1 | 68.3 (0.0) | Tabpred:lm_KNeighborsUnif_BAG_L1 | 41.8 (0.0) |
| Tabpred:dtm_ExtraTreesGini_BAG_L1 | 68.3 (0.0) | Tabpred:lm_KNeighborsDist_BAG_L1 | 41.7 (0.0) |
| Tabpred:dtm_LightGBMXT_BAG_L1 | 68.3 (0.0) | Tabpred:dtm_KNeighborsUnif_BAG_L1 | 40.2 (0.0) |
| Tabpred:dtm_NeuralNetFastAI_BAG_L1 | 68.0 (0.0) | LMdict-lin | 37.3 (nan) |
| WordCount-lin | 67.8 (1.1) | LMdict-naive | 29.6 (nan) |
| Tabpred:dtm_ExtraTreesEntr_BAG_L1 | 67.0 (0.0) | LDA-K100 | 26.4 (0.2) |
| Tabpred:dtm_RandomForestEntr_BAG_L1 | 66.8 (0.0) | LDA-K5 | 26.4 (0.0) |
| Tabpred:dtm_NeuralNetTorch_BAG_L2 | 57.9 (0.0) | LDA-K10 | 26.4 (0.0) |
| Tabpred:lm_NeuralNetFastAI_BAG_L2 | 51.5 (0.0) | LDA-K50 | 26.4 (0.0) |
| Tabpred:lm_NeuralNetFastAI_BAG_L1 | 50.1 (0.0) | LDA-K500 | 26.4 (0.0) |
| Tabpred:lm_WeightedEnsemble_L2 | 50.0 (0.0) | LDA-K1000 | 26.4 (0.0) |
| Tabpred:lm_NeuralNetTorch_BAG_L2 | 48.5 (0.0) | majority class | 26.4 (0.0) |
| Tabpred:lm_CatBoost_BAG_L2 | 46.7 (0.0) | Tabpred:dtm_NeuralNetTorch_BAG_L1 | 26.4 (0.0) |
| Tabpred:lm_XGBoost_BAG_L2 | 46.4 (0.0) | LDA-K250 | 26.4 (0.0) |

Avg. over 50 runs, standard deviation in brackets, train size: 9,545, test size: 2,386 (not shuffled)

### A.2.3 Financial Phrase Bank

Table A.3: Financial Phrase Bank benchmark

| model | macro F1 | model | macro F1 |
|---|---|---|---|
| FinBERT* | 95.0 (-) | Tabpred:lm_CatBoost_BAG_L2 | 56.9 (2.2) |
| BERT-base | 94.4 (1.5) | Tabpred:lm_LightGBMXT_BAG_L1 | 56.6 (2.2) |
| FLANG-BERT | 94.1 (1.5) | Tabpred:lm_CatBoost_BAG_L1 | 56.6 (2.1) |
| Tabpred:dtm_LightGBMXT_BAG_L2 | 87.1 (1.6) | Tabpred:lm_RandomForestGini_BAG_L2 | 56.4 (2.1) |
| Tabpred:dtm_WeightedEnsemble_L3 | 87.0 (1.6) | Tabpred:lm_RandomForestEntr_BAG_L2 | 56.1 (2.4) |
| Tabpred:dtm_LightGBMLarge_BAG_L2 | 86.9 (1.6) | Tabpred:lm_ExtraTreesEntr_BAG_L2 | 56.1 (2.3) |
| Tabpred:dtm_LightGBM_BAG_L2 | 86.8 (1.5) | Tabpred:lm_WeightedEnsemble_L3 | 56.1 (2.3) |
| Tabpred:dtm_XGBoost_BAG_L2 | 86.8 (1.6) | Tabpred:lm_ExtraTreesGini_BAG_L2 | 56.0 (2.5) |
| Tabpred:dtm_CatBoost_BAG_L2 | 86.8 (1.6) | Tabpred:lm_XGBoost_BAG_L2 | 55.7 (2.0) |
| Tabpred:dtm_RandomForestEntr_BAG_L2 | 85.9 (1.8) | Tabpred:lm_LightGBMXT_BAG_L2 | 55.6 (2.1) |
| Tabpred:dtm_RandomForestGini_BAG_L2 | 85.8 (1.9) | Tabpred:lm_LightGBMLarge_BAG_L2 | 55.5 (2.0) |
| Tabpred:dtm_ExtraTreesGini_BAG_L2 | 85.6 (1.8) | Tabpred:lm_LightGBM_BAG_L2 | 55.4 (2.2) |
| Tabpred:dtm_ExtraTreesEntr_BAG_L2 | 85.5 (1.8) | Tabpred:lm_XGBoost_BAG_L1 | 55.3 (2.1) |
| Tabpred:dtm_NeuralNetFastAI_BAG_L2 | 85.3 (1.6) | Tabpred:lm_LightGBM_BAG_L1 | 54.4 (2.3) |
| Tabpred:dtm_CatBoost_BAG_L1 | 85.2 (1.9) | Tabpred:lm_ExtraTreesGini_BAG_L1 | 54.3 (2.1) |
| Tabpred:dtm_NeuralNetTorch_BAG_L2 | 85.1 (1.8) | Tabpred:lm_ExtraTreesEntr_BAG_L1 | 54.2 (2.2) |
| Tabpred:dtm_WeightedEnsemble_L2 | 84.9 (1.9) | Tabpred:lm_RandomForestEntr_BAG_L1 | 53.8 (2.2) |
| Tabpred:dtm_XGBoost_BAG_L1 | 83.5 (2.0) | Tabpred:lm_RandomForestGini_BAG_L1 | 53.7 (2.2) |
| Tabpred:dtm_LightGBMLarge_BAG_L1 | 83.3 (1.8) | Tabpred:lm_LightGBMLarge_BAG_L1 | 52.9 (2.1) |
| WordCount-lin | 82.9 (1.8) | Tabpred:dtm_KNeighborsDist_BAG_L1 | 52.8 (3.1) |
| Tabpred:dtm_LightGBM_BAG_L1 | 79.4 (2.6) | Tabpred:lm_KNeighborsDist_BAG_L1 | 52.2 (2.3) |
| Tabpred:dtm_NeuralNetFastAI_BAG_L1 | 78.8 (2.3) | Tabpred:dtm_NeuralNetTorch_BAG_L1 | 52.1 (1.2) |
| Tabpred:dtm_LightGBMXT_BAG_L1 | 78.8 (2.7) | Tabpred:lm_KNeighborsUnif_BAG_L1 | 52.0 (1.9) |
| Tabpred:dtm_ExtraTreesGini_BAG_L1 | 75.6 (2.5) | Tabpred:dtm_KNeighborsUnif_BAG_L1 | 51.8 (2.9) |
| Tabpred:dtm_ExtraTreesEntr_BAG_L1 | 74.8 (2.7) | LMdict-lin | 50.0 (0.0) |
| Tabpred:dtm_RandomForestGini_BAG_L1 | 74.7 (2.8) | LDA-K500 | 47.6 (9.3) |
| Tabpred:dtm_RandomForestEntr_BAG_L1 | 74.2 (2.7) | LDA-K100 | 44.1 (4.0) |
| LDA-K1000 | 63.7 (4.0) | LDA-K250 | 44.0 (4.4) |
| Tabpred:lm_NeuralNetFastAI_BAG_L1 | 58.3 (2.3) | LDA-K50 | 42.7 (3.2) |
| Tabpred:lm_WeightedEnsemble_L2 | 58.3 (2.2) | LDA-K10 | 29.6 (6.1) |
| Tabpred:lm_NeuralNetTorch_BAG_L2 | 57.8 (2.1) | LDA-K5 | 25.6 (2.2) |
| Tabpred:lm_NeuralNetFastAI_BAG_L2 | 57.7 (2.3) | majority class | 25.1 (nan) |
| Tabpred:lm_NeuralNetTorch_BAG_L1 | 57.2 (2.2) | LMdict-naive | 23.0 (0.0) |

Avg. over 50 runs, standard deviation in brackets, train size: 1,698, test size: 566. FinBERT was trained on part of the Financial Phrase Bank. We report the F1-score of the test set performance reported in the original paper Araci (2019). When we fine-tuned FinBERT on the Financial Phrase Bank nonetheless, we measured an F1 score of 94.7 (1.2) on our test set. Note that there might data leakage between our test set and the original FinBERT test set.

## A.3 Multimodal Datasets - Detailed Results

### A.3.1 FOMC Greenbook CPI forecast

Table A.4: FOMC Greenbook CPI Multimodal

| model | macro F1 | model | macro F1 |
|---|---|---|---|
| Tabpred:Tab+lm_ExtraTreesGini_BAG_L1 | 46.7 (0.0) | textpred:Tab+text_WeightedEnsemble_L2 | 34.8 (0.0) |
| Tabpred:Tab+lm_XGBoost_BAG_L1 | 45.2 (0.0) | textpred:Tab+text_LightGBMXT_BAG_L1 | 34.8 (0.0) |
| Tabpred:tab_LightGBMXT_BAG_L1 | 44.2 (0.0) | Tabpred:dtm_KNeighborsDist_BAG_L1 | 34.8 (0.0) |
| Tabpred:Tab+lm_NeuralNetFastAI_BAG_L1 | 43.6 (0.0) | Tabpred:Tab+dtm_LightGBMLarge_BAG_L1 | 34.8 (0.0) |
| textpred:Tab+text_NeuralNetTorch_BAG_L1 | 43.1 (0.0) | textpred:Tab+text_XGBoost_BAG_L1 | 34.8 (0.0) |
| Tabpred:Tab+dtm_WeightedEnsemble_L2 | 42.8 (0.0) | Tabpred:lm_XGBoost_BAG_L1 | 34.8 (0.0) |
| Tabpred:Tab+dtm_XGBoost_BAG_L1 | 42.7 (0.0) | Tabpred:dtm_KNeighborsUnif_BAG_L1 | 34.8 (0.0) |
| Tabpred:tab_CatBoost_BAG_L1 | 42.1 (0.0) | Tabpred:lm_ExtraTreesEntr_BAG_L1 | 34.7 (0.0) |
| Tabpred:Tab+lm_LightGBMLarge_BAG_L1 | 42.1 (0.0) | Tabpred:lm_RandomForestGini_BAG_L1 | 34.1 (0.0) |
| Tabpred:dtm_LightGBMXT_BAG_L1 | 42.1 (0.0) | Tabpred:Tab+dtm_NeuralNetFastAI_BAG_L1 | 33.9 (0.0) |
| Tabpred:tab_NeuralNetFastAI_BAG_L1 | 41.7 (0.0) | Tabpred:dtm_XGBoost_BAG_L1 | 32.8 (0.0) |
| Tabpred:tab_LightGBMLarge_BAG_L1 | 41.6 (0.0) | textpred:Tab+text_LightGBMLarge_BAG_L1 | 32.8 (0.0) |
| Tabpred:Tab+dtm_RandomForestGini_BAG_L1 | 40.9 (0.0) | Tabpred:lm_NeuralNetFastAI_BAG_L1 | 32.1 (0.0) |
| Tabpred:tab_ExtraTreesEntr_BAG_L1 | 39.8 (0.0) | Tabpred:lm_LightGBMLarge_BAG_L1 | 32.1 (0.0) |
| Tabpred:Tab+lm_WeightedEnsemble_L2 | 39.8 (0.0) | Tabpred:dtm_CatBoost_BAG_L1 | 32.1 (0.0) |
| Tabpred:Tab+lm_LightGBMXT_BAG_L1 | 39.8 (0.0) | textpred:Tab+text_LightGBM_BAG_L1 | 32.1 (0.0) |
| Tabpred:Tab+lm_ExtraTreesEntr_BAG_L1 | 39.8 (0.0) | BERT-base | 31.0 (7.1) |
| Tabpred:tab_ExtraTreesGini_BAG_L1 | 39.7 (0.0) | LDA-K100 | 30.6 (8.3) |
| Tabpred:tab_XGBoost_BAG_L1 | 39.7 (0.0) | LDA-K50 | 30.3 (8.0) |
| Tabpred:Tab+dtm_RandomForestEntr_BAG_L1 | 39.5 (0.0) | Tabpred:lm_CatBoost_BAG_L1 | 30.0 (0.0) |
| Tabpred:dtm_LightGBMLarge_BAG_L1 | 39.5 (0.0) | Tabpred:dtm_LightGBM_BAG_L1 | 30.0 (0.0) |
| Tabpred:Tab+dtm_CatBoost_BAG_L1 | 38.9 (0.0) | LDA-K10 | 29.9 (7.9) |
| textpred:Tab+text_CatBoost_BAG_L1 | 38.9 (0.0) | LDA-K250 | 29.9 (7.6) |
| Tabpred:Tab+dtm_ExtraTreesGini_BAG_L1 | 38.9 (0.0) | LDA-K1000 | 29.5 (5.6) |
| Tabpred:dtm_ExtraTreesGini_BAG_L1 | 38.0 (0.0) | LDA-K500 | 29.5 (6.3) |
| Tabpred:Tab+lm_NeuralNetTorch_BAG_L1 | 37.4 (0.0) | FinBERT | 29.0 (7.2) |
| Tabpred:tab_WeightedEnsemble_L2 | 37.3 (0.0) | FLANG-BERT | 28.7 (6.6) |
| Tabpred:tab_LightGBM_BAG_L1 | 37.3 (0.0) | MMpred:Tab+text_MMpredictor | 28.1 (0.0) |
| Tabpred:Tab+dtm_NeuralNetTorch_BAG_L1 | 37.3 (0.0) | LDA-K5 | 28.1 (7.6) |
| Tabpred:Tab+lm_LightGBM_BAG_L1 | 37.3 (0.0) | Tabpred:lm_NeuralNetTorch_BAG_L1 | 27.3 (0.0) |
| Tabpred:Tab+lm_CatBoost_BAG_L1 | 37.3 (0.0) | Tabpred:lm_WeightedEnsemble_L2 | 27.3 (0.0) |
| Tabpred:tab_RandomForestEntr_BAG_L1 | 37.3 (0.0) | WordCount-lin | 25.6 (1.9) |
| Tabpred:Tab+dtm_LightGBM_BAG_L1 | 37.3 (0.0) | Tabpred:lm_KNeighborsDist_BAG_L1 | 23.6 (0.0) |
| Tabpred:Tab+lm_RandomForestEntr_BAG_L1 | 37.3 (0.0) | Tabpred:lm_LightGBM_BAG_L1 | 23.4 (0.0) |
| Tabpred:lm_ExtraTreesGini_BAG_L1 | 37.3 (0.0) | LMdict-lin | 23.3 (0.0) |
| Tabpred:Tab+dtm_LightGBMXT_BAG_L1 | 37.3 (0.0) | Tabpred:Tab+lm_KNeighborsUnif_BAG_L1 | 21.7 (0.0) |
| Tabpred:dtm_NeuralNetFastAI_BAG_L1 | 36.9 (0.0) | Tabpred:tab_KNeighborsUnif_BAG_L1 | 21.7 (0.0) |
| Tabpred:tab_NeuralNetTorch_BAG_L1 | 36.9 (0.0) | majority class | 21.7 (nan) |
| Tabpred:lm_RandomForestEntr_BAG_L1 | 36.9 (0.0) | Tabpred:Tab+dtm_KNeighborsUnif_BAG_L1 | 21.7 (0.0) |
| Tabpred:lm_KNeighborsUnif_BAG_L1 | 36.9 (0.0) | Tabpred:Tab+dtm_KNeighborsDist_BAG_L1 | 21.7 (0.0) |
| Tabpred:dtm_WeightedEnsemble_L2 | 36.8 (0.0) | Tabpred:Tab+lm_KNeighborsDist_BAG_L1 | 21.7 (0.0) |
| Tabpred:tab_RandomForestGini_BAG_L1 | 36.1 (0.0) | Tabpred:tab_KNeighborsDist_BAG_L1 | 21.7 (0.0) |
| Tabpred:dtm_ExtraTreesEntr_BAG_L1 | 36.1 (0.0) | textpred:Tab+text_TextPredictor_BAG_L1 | 21.7 (0.0) |
| Tabpred:dtm_RandomForestEntr_BAG_L1 | 36.1 (0.0) | Tabpred:lm_LightGBMXT_BAG_L1 | 21.5 (0.0) |
| Tabpred:Tab+dtm_ExtraTreesEntr_BAG_L1 | 36.1 (0.0) | Tabpred:dtm_NeuralNetTorch_BAG_L1 | 21.1 (0.0) |
| Tabpred:dtm_RandomForestGini_BAG_L1 | 36.1 (0.0) | LMdict-naive | 15.3 (0.0) |
| Tabpred:Tab+lm_RandomForestGini_BAG_L1 | 35.1 (0.0) | | |

Avg. over 50 runs, standard deviation in brackets, #train: 112, #test: 28

# B

# Appendix: Bayesian Topic Regression

## B.1 Observations without Documents

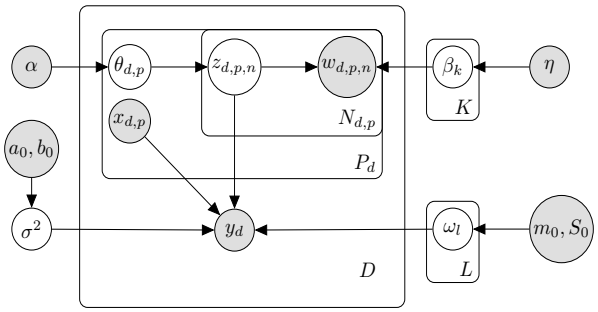
A straightforward extension allows for some observations to be associated with an $\boldsymbol{X}$ and $\boldsymbol{y}$, but no document. This is often the case in a social science context, for example time-series may be associated with documents at irregular intervals. If an observation is not associated with any documents, the priors on the document topic distributions suggest that the topic assignment for topic $K$ is set to $\alpha_k / \sum_k \alpha_k$. These observations may still be very useful in estimating the relationship between $\boldsymbol{X}$ and $\boldsymbol{y}$ so they are worth including in the estimation.

### B.1.1 Multiple Paragraphs

If, as is often the case in the context of social science applications, we have relatively few observations but the documents associated with those observations are relatively long, we can exploit the structure of the documents by estimating the model at a paragraph level. Splitting up longer documents into paragraphs brings one of the key advantages of topic modelling to the fore: that the same word can have different meanings in different contexts. For example, the word "increase" might have quite a different meaning if it is in a paragraph with the word "risk" than if it is alongside "productivity". Treating the entire document as a single bag of words makes it hard for the model to make this distinction.

If there are observations with multiple documents, we can treat these as $P_d$ separate paragraphs of a combined document, indexed by $p$, each with an independent $\boldsymbol{\theta}_p$ distribution over topics. These paragraphs may also have different associated $\boldsymbol{x}_{d,p}$ that interact with the topics, for example we may wish to interact topics with a paragraph specific sentiment score, but the

Figure B.1: Graphical model for BTR with multiple documents per observation




response variable $y_d$ is common to all paragraphs in the same document and the M-step estimated at the document level. Figure B.1 shows the extended graphical model.

If $\boldsymbol{x_{d,p}}$ only enters linearly into the regression then some document-level average will have to be used and this transformation can be performed prior to estimation, converting it into an $\boldsymbol{x}_{1,d}$, and so the algorithm will remain unchanged. However, if any of the $\boldsymbol{x}_{d,p}$ variables are interacted with $\bar{\boldsymbol{z}}_{d,p}$ then we may wish for this interaction to be at the paragraph level. For example, if we think that a topic might have a different effect depending on the sentiment of the surrounding paragraph. In this case, we still need to aggregate the interaction to the document level, but aggregate after interacting rather than interacting after aggregating. We therefore define

$$\overline{\boldsymbol{x}_{d,p} \otimes \boldsymbol{z}_{d,p}} = \frac{1}{N_d} \sum_{p \in [P_d]} \sum_{n \in [N_{d,p}]} [\boldsymbol{x}_{d,p} \otimes \boldsymbol{s}_{d,p,n}] \tag{B.1}$$

where $[N]$ denotes the set of integers $\{1, ..., N\}$ and $\otimes$ represents the Kronecker product. The design matrix $\boldsymbol{A}$ is then

$$\boldsymbol{A} = \begin{bmatrix} \bar{\boldsymbol{z}}_1 & \overline{\boldsymbol{x}_{1,1,p} \otimes \boldsymbol{z}_{1,p}} & \boldsymbol{x}_{2,1} \\ \vdots & \vdots & \vdots \\ \bar{\boldsymbol{z}}_1 & \overline{\boldsymbol{x}_{1,d,p} \otimes \boldsymbol{z}_{d,p}} & \boldsymbol{x}_{2,d} \\ \vdots & \vdots & \vdots \\ \bar{\boldsymbol{z}}_1 & \overline{\boldsymbol{x}_{1,D,p} \otimes \boldsymbol{z}_{D,p}} & \boldsymbol{x}_{2,D} \end{bmatrix} \tag{B.2}$$

and the predictive model for $y_d$ will be

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{A}\boldsymbol{\omega}, \sigma^2) \quad \text{where} \quad \boldsymbol{\omega} = (\boldsymbol{\omega}_z, \boldsymbol{\omega}_{zx}, \boldsymbol{\omega}_x). \tag{B.3}$$

The simplest way to aggregate from paragraphs to documents is simply to give each word in the document equal weight as above. This will mean that longer paragraphs have greater weight than shorter ones.

As before, we can collapse out the latent variables $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ so that we only need to sample for the topic assignments $\boldsymbol{z}$ in an E-step and then for $\boldsymbol{\omega}$ and $\sigma^2$ in an M-step.

In the E-step, we need to sample from the conditional posterior for the topic assignment of each word

$$\Pr[z_{d,p,n} = k | \boldsymbol{Z}_{d,-(p,n)}, \boldsymbol{W}, \alpha, \eta, \boldsymbol{y}, \boldsymbol{X}, \boldsymbol{\omega}, \sigma^2]. \tag{B.4}$$

By the conditional independence properties of the graphical model, we can split this into $p(\boldsymbol{Z}|\boldsymbol{W}, \alpha, \eta)$ and $p(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{\omega}, \sigma^2)$. The sampling equation for the $n$th token in the $p$th paragraph of the $d$th document $d$ will have the form

$$\begin{aligned} \Pr[z_{d,p,n} = k | \boldsymbol{Z}_{d,-(p,n)}, \boldsymbol{W}, \alpha, \eta, \boldsymbol{y}, \boldsymbol{X}, \boldsymbol{\omega}, \sigma^2] \propto \\ \Pr[z_{d,p,n} = k | \boldsymbol{Z}_{d,p,-(n)}, \boldsymbol{W}, \alpha, \eta] \quad \times \quad \Pr[y_d | z_{d,p,n} = k, \boldsymbol{Z}_{d,-(p,n)}, \boldsymbol{x}_d, \omega, \sigma^2]. \end{aligned} \tag{B.5}$$

The topic assignment each document is independent, but there are dependencies across paragraphs. Crucially, these paragraphs have are independent with respect to $\boldsymbol{\theta}$, so $p(\boldsymbol{Z}|\boldsymbol{W}, \alpha, \eta)$ is paragraph specific.

$$\Pr[z_{d,p,n} = k | \boldsymbol{Z}_{d,p-(n)}, \boldsymbol{W}, \alpha, \eta] \propto (s_{d,p,k,-n} + \alpha) \frac{m_{k,v,-(d,p,n)} + \eta}{\sum_v m_{k,v,-(d,p,n)} + V\eta}. \tag{B.6}$$

However, the regression part is at the document level to $p(\boldsymbol{y}|\boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{\omega}, \sigma^2)$ will condition on all the paragraphs in a given document. Given that the residuals are Gaussian, the probability of the outcome variable for a given document $d$ is

$$p(\boldsymbol{y}_d | \boldsymbol{z}_d, \boldsymbol{x}_d, \boldsymbol{\omega}, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[ -\frac{(y_d - \boldsymbol{\omega}_z^\intercal \bar{\boldsymbol{z}}_d - \boldsymbol{\omega}_{zx}^\intercal (\overline{\boldsymbol{x}_{1,d,p} \otimes \boldsymbol{z}_{d,p}}) - \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_{2,d})^2}{2\sigma^2} \right]. \tag{B.7}$$

We can write this in a convenient form that preserves proportionality with respect to $z_{d,p,n}$

such that it depends only on the data and count variables used in the other two terms and the document-wide counts. First we can break the prediction for $y_d$ into the section that depends on paragraph $p$ and the section that depends on other paragraphs and document wide $\boldsymbol{x}_{1,d}$.

$$y_d - \boldsymbol{\omega}_z^\intercal \bar{\boldsymbol{z}}_d - \boldsymbol{\omega}_{zx}^\intercal (\overline{\boldsymbol{x}_{1,d,p} \otimes \boldsymbol{z}_{d,p}}) = \left( y_d - \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_{2,d} - \frac{\boldsymbol{\omega}_z^\intercal}{N_d} \boldsymbol{s}_{d,-p} - \frac{\boldsymbol{\omega}_{zx}^\intercal}{N_d} \sum_{q \in \{[P_d] \backslash p\}} [\boldsymbol{x}_{1,d,q} \otimes \boldsymbol{s}_{d,q}] \right) - \left( \frac{\boldsymbol{\omega}_z^\intercal}{N_d} (\boldsymbol{s}_{d,p,-n} + \boldsymbol{s}_{d,p,n}) - \frac{\boldsymbol{\omega}_{zx}^\intercal}{N_d} \boldsymbol{x}_{1,d,p} \otimes (\boldsymbol{s}_{d,p,-n} + \boldsymbol{s}_{d,p,n}) \right) \tag{B.8}$$

where $N_d$ is the total number of words in the *document*.

Define $\hat{y}_{d,-p}$ as the predicted $y_d$ without paragraph $p$,

$$\hat{y}_{d,-p} = \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_{2,d} + \frac{\boldsymbol{\omega}_z^\intercal}{N_d} \boldsymbol{s}_{d,-p} + \frac{\boldsymbol{\omega}_{zx}^\intercal}{N_d} \sum_{q \in \{[P_d] \backslash p\}} [\boldsymbol{x}_{1,d,q} \otimes \boldsymbol{s}_{d,q}] \,. \tag{B.9}$$

We then have a predictive distribution that depends only on paragraph $p$.

$$y_d \sim \mathcal{N} \left( \hat{y}_{d,-p} - \frac{\boldsymbol{\omega}_z^\intercal}{N_d} (\boldsymbol{s}_{d,p,-n} + \boldsymbol{s}_{d,p,n}) - \frac{\boldsymbol{\omega}'_{zx}}{N_d} \boldsymbol{x}_{1,d,p} \otimes (\boldsymbol{s}_{d,p,-n} + \boldsymbol{s}_{d,p,n}), \sigma^2 \right) . \tag{B.10}$$

We can then follow the same steps as for the single paragraph document case to derive the third term in the sampling distribution, defining $\tilde{\boldsymbol{\omega}}_{z,d,p,k} = \boldsymbol{\omega}_{z,k} + \boldsymbol{\omega}'_{zx,k} \boldsymbol{x}_{1,d,p}$ analogously to $\tilde{\boldsymbol{\omega}}$ defined for the single paragraph case.

This gives us the sampling distribution for $z$, which is a Multinomial parameterised by

$$\Pr[z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}, \boldsymbol{y}, \alpha, \eta, \boldsymbol{\omega}, \sigma^2] \propto (s_{d,p,k,-n} + \alpha) \frac{m_{k,v,-(d,p,n)} + \eta}{\sum_v m_{k,v,-(d,p,n)} + V\eta} \exp \left[ \frac{1}{2\sigma^2} \left( \frac{2\tilde{\omega}_{z,d,p,k}}{N_d} \left( y_d - \hat{y}_{d,-p} - \frac{\tilde{\boldsymbol{\omega}}'_{z,d,p}}{N_d} \boldsymbol{s}_{d,-n} \right) - \left( \frac{\tilde{\omega}_{z,d,p,k}}{N_d} \right) \right)^2 \right] . \tag{B.11}$$

In the M-step we can then still use the average $\bar{z}_{d,p}$ estimated in the E-step, but we need to weight each paragraph by the number of words in that paragraph to be consistent with the

E-step,

$$\bar{z}_d = \frac{1}{N_d} \sum_{p \in [P_d]} [N_{d,p} \bar{z}_{d,p}] \tag{B.12}$$

$$(\overline{x_{1,d,p} \otimes z_{d,p}}) = \frac{1}{N_d} \sum_{p \in [P_d]} [N_{d,p} x_{1,d,p} \otimes z_{d,p}] \,. \tag{B.13}$$

## B.2 Gibbs-EM Algorithm

### B.2.1 Sampling Distribution for $z$

The probability of a given word $w_{d,n}$ being assigned to a given topic $k$ (such that $z_{d,n} = k$), conditional on the assignments of all other words (as well as the model's other latent variables and the data) is

$$p(z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2), \tag{B.14}$$

where $\boldsymbol{Z}_{-(d,n)}$ are the topic assignments for all words apart from $w_{d,n}$. By the conditional independence properties implied by the graphical model, we can split this joint posterior into

$$p(\boldsymbol{Z} | \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2) \propto p(\boldsymbol{Z} | \boldsymbol{W}) p(\boldsymbol{y} | \boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{\omega}, \sigma^2). \tag{B.15}$$

As topic assignments within one document are independent from topic assignments in all other documents, the sampling equation for the $n$th word in document $d$ should only depend it's own response variable, $y_d$, such that

$$p(z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \boldsymbol{\omega}, \sigma^2) \propto p(z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}) p(y_d | z_{d,n} = k, \boldsymbol{Z}_{-(d,n)}, \boldsymbol{x}_d, \boldsymbol{\omega}, \sigma^2). \tag{B.16}$$

The first part of the RHS expression is just the sampling distribution of a standard LDA model, so it can be expressed in terms of the count variables $\boldsymbol{s}$ (the topic assignments across a document) and $\boldsymbol{m}$ (the assignments of unique words across topics over all documents). $s_{d,k}$ measures the total number of words in document $d$ assigned to topic $k$ and $s_{d,k,-n}$ the number of words in

document $d$ assigned to topic $k$, except for word $n$. Analogously, $m_{k,v}$ measures the total number of times term $v$ is assigned to topic $k$ across all documents and $m_{k,v,-(d,n)}$ measures the same, but excludes word $n$ in document $d$.

$$p(z_{d,n} = k|\boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}) \propto (s_{d,k,-n} + \alpha)\frac{m_{k,v,-(d,n)} + \eta}{\sum_v m_{k,v,-(d,n)} + V\eta}. \tag{B.17}$$

#### B.2.1.1 Regression

Given that the residuals are Gaussian, the probability of the response variable for a given document $d$ is

$$p(y_d|\boldsymbol{z}_d, \boldsymbol{x}_d, \boldsymbol{\omega}, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(y_d - \boldsymbol{\omega}^\intercal \boldsymbol{a}_d)^2}{2\sigma^2}\right\}. \tag{B.18}$$

We can write this in a convenient form that preserves proportionality with respect to $z_{d,n}$ such that it depends only on the data and count variables used in the other two terms. First, we split the $\boldsymbol{x}_d$ features into those that are interacted, $\boldsymbol{x}_{1,d}$, and those that are not, $\boldsymbol{x}_{2,d}$. The generative model for $y_d$ is then

$$y_d \sim \mathcal{N}(\boldsymbol{\omega}_z^\intercal \bar{\boldsymbol{z}}_d + \boldsymbol{\omega}_{zx}^\intercal(\boldsymbol{x}_{1,d} \otimes \bar{\boldsymbol{z}}_d) + \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_{2,d}, \sigma^2). \tag{B.19}$$

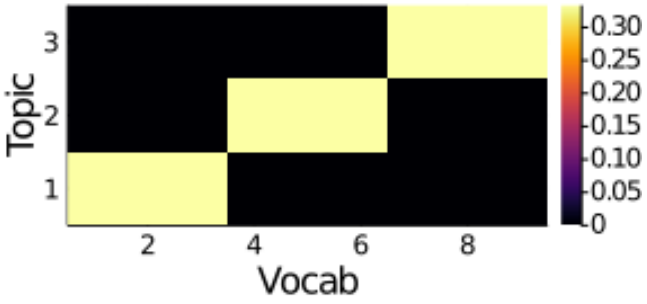
where $\otimes$ is the Kronecker product. Noting that $\boldsymbol{X}$ is observed, so we can think of this as a linear model with document-specific regression parameters. Define $\tilde{\boldsymbol{\omega}}_{z,d}$ as a length $K$ vector such that

$$\tilde{\omega}_{z,d,k} = \omega_{z,k} + \boldsymbol{\omega}_{zx,k}^\intercal \boldsymbol{x}_{1,d}. \tag{B.20}$$

Noting that $\tilde{\boldsymbol{\omega}}_{z,d}^\intercal \bar{\boldsymbol{z}}_d = \frac{\tilde{\omega}_{\boldsymbol{z},\boldsymbol{d}}{}^\intercal}{N_d}(\boldsymbol{s}_{d,-n} + \boldsymbol{s}_{d,n})$, the probability density of $y$ conditional on $z_{d,n} = k$ is therefore proportional to

$$\begin{aligned} &p(y_d|z_{d,n} = k, \boldsymbol{z}_{-(d,n)}, \boldsymbol{x}_d, \boldsymbol{\omega}, \sigma^2) \propto \\ &\exp\left\{\frac{1}{2\sigma^2}\left(\frac{2\tilde{\omega}_{z,d,k}}{N_d}\left(y_d - \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_d - \frac{\tilde{\boldsymbol{\omega}}_{z,d}^\intercal}{N_d}\boldsymbol{s}_{d,-n}\right) - \left(\frac{\tilde{\omega}_{z,d,k}}{N_d}\right)^2\right)\right\}. \end{aligned} \tag{B.21}$$

Figure B.2: Ground truth topic distribution for synthetic documents.




This gives us the sampling distribution for $z_{d,n}$ stated in equation (B.16): a multinomial distribution parameterised by

$$p(z_{d,n} = k | \boldsymbol{Z}_{-(d,n)}, \boldsymbol{W}, \boldsymbol{X}, \boldsymbol{y}, \alpha, \eta, \boldsymbol{\omega}, \sigma^2) \propto$$
$$(s_{d,k,-n} + \alpha) \frac{m_{k,v,-(d,n)} + \eta}{\sum_v m_{k,v,-(d,n)} + V\eta}$$
$$\exp\left\{ \frac{1}{2\sigma^2} \left( \frac{2\tilde{\omega}_{z,d,k}}{N_d} \left( y_d - \boldsymbol{\omega}_x^\intercal \boldsymbol{x}_{2,d} - \frac{\tilde{\boldsymbol{\omega}}_{z,d}^\intercal}{N_d} \boldsymbol{s}_{d,-n} \right) - \left( \frac{\tilde{\omega}_{z,d,k}}{N_d} \right)^2 \right) \right\}. \tag{B.22}$$

This defines for each $k \in \{1, ..., K\}$ the probability that $z_{d,n}$ is assigned to that topic. These $K$ probabilities define the multinomial distribution from which $z_{d,n}$ is drawn.

#### B.2.1.2 $\theta$ and $\beta$

Given topic assignments $z$, we can recover the latent variables $\theta$ and $\beta$ from their predictive distributions via

$$\hat{\theta}_{d,k} = \frac{s_{d,k} + \alpha}{\sum_k (s_{d,k} + \alpha)} \tag{B.23}$$

and

$$\hat{\beta}_{k,v} = \frac{m_{k,v} + \eta}{\sum_v (m_{k,v} + \eta)}. \tag{B.24}$$

## B.3 Synthetic Data Experiments

Figure B.2 shows the topic-vocabulary distribution from which the synthetic documents are generated.

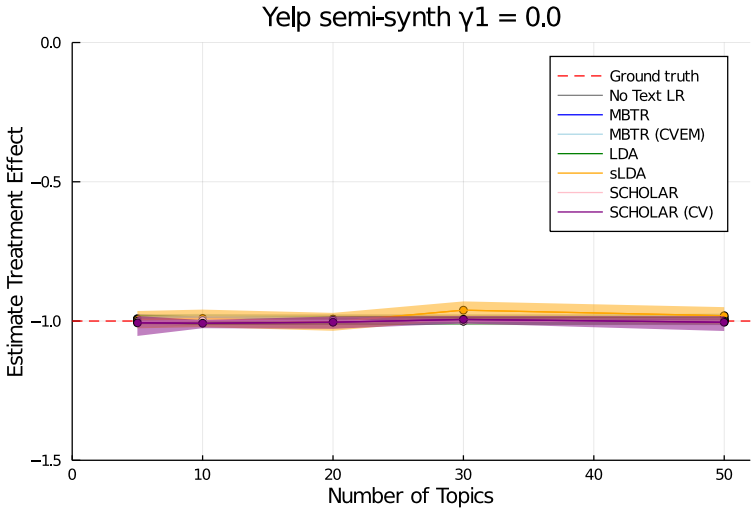
Table B.1 shows the hyperparameter settings used in the synthetic data section. We observed that the settings of the prior did hardly effect results, given the strong signal in the synthetic dataset.

Table B.1: Synthetic example hyperparameters

| | K | $\alpha$ | $\eta$ | $\mu_{\text{ntm}}$ | $\sigma_{\text{ntm}}$ | $a_0$ | $b_0$ | $m_0$ | $S_0$ |
|---|---|---|---|---|---|---|---|---|---|
| LDA | 3 | 1.0 | 1.0 | - | - | - | - | - | - |
| sLDA | 3 | 1.0 | 1.0 | - | - | - | - | - | - |
| BPsLDA | 3 | 1.0 | 1.0 | - | - | - | - | - | - |
| BTR | 3 | 1.0 | 1.0 | - | - | 0.2 | 4 | 0 | 2 |

## B.4 Semi-Synthetic Data Experiments

Figure B.3: No correlation between confounders and treatments




Without correlation between confounders and treatments, the regression can be dissected into two separate parts (supervised topic estimation and regression weight estimation of the non-text features) without inducing bias in the estimators, as described in the section on the Frisch-Waugh-Lovell theorem. In such a case, all models manage to recover the ground truth.

## B.5 Real-World Datasets and Data Pre-Processing

The **Yelp dataset** contains over 8 million customer reviews of businesses, which we restrict to reviews for businesses in Toronto. The **Booking dataset** contains around $500,000$ hotel reviews. For both datasets, we randomly sample $50,000$ observations and randomly select 75% in Yelp, 80% in Booking of our sample for training, holding out the remainder for testing. We then further split the training set equally for training in the E-step and validation in the

M-step. The features are normalized on the training data statistics and the response variable is de-meaned. We do this because the $K$ topic features sum to one and therefore implicitly already add a constant to the regression (Blei and McAuliffe, 2008). We preprocess the text corpora by removing stopwords and then tokenizing and stemming the data.

Table B.2: Summary statistics of the review datasets

| **Statistics** | #train | #val | #test | #vocab | #max words | #avg words |
|---|---|---|---|---|---|---|
| **Yelp** | 18,750 | 18,750 | 12,500 | 24,680 | 572 | 61.2 |
| **Booking** | 20,000 | 20,000 | 10,000 | 6,968 | 305 | 18.7 |

The Booking.com dataset allows consumers to enter the positive and negative parts of their reviews in separate boxes. We combine these two reviews for all our exercises, but we do use information on the word count in each of these sections (see below).

For the prediction exercises in Section 5.8, we use the number of stars associated with each review as the target variable. We also use the numerical metadata described in Table B.3as covariates.

Table B.3: Numerical covariates for prediction experiments

| **Dataset** | Variable | Description |
|---|---|---|
| 3cYelp | stars_av_u | historic avg. rating by user |
| | stars_av_b | historic avg. rating of business |
| | sentiment | *Harvard Inquirer* sentiment score |
| 3cBooking | Average$_S core$ | historical averge hotel score |
| | Review_Total_Negative_Word_Counts | total number of words in the negative part of review |
| | Review_Total_Positive_Word_Counts | total number of words in the positive part of review |
| | Total_Number_of_Reviews_Reviewer_Has_Given | total num of reviews by customer |
| | Total_Number_of_Reviews | total num of reviews of hotel |

For the semi-synthetic exercise on the Booking data, we construct

$$pos\_prop_i = \frac{Review\_Total\_Positive\_Word\_Counts_i}{Review\_Total\_Positive\_Word\_Counts_i + Review\_Total\_Negative\_Word\_Counts_i}$$

This variable is correlated with the treatment ($Average\_Score_i$) and with the outcome, and so the text can act as a confounder.

## B.6 Real-World Data Experiments

### B.6.1 Empirical data evaluation across different K

Table B.4: Mean $pR^2$ and perplexity

| *Dataset* | Booking | | | | Yelp | | | |
|---|---|---|---|---|---|---|---|---|
| *K* | 10 | 20 | 30 | 50 | 10 | 20 | 30 | 50 |
| | | | | $pR^2$ (higher is better) | | | | |
| LDA+LR | 0.400 (0.003) | 0.410 (0.004) | 0.417 (0.005) | 0.426 (0.003) | 0.498 (0.005) | 0.530 (0.009) | 0.561 (0.010) | 0.586 (0.006) |
| GSM+LR | 0.387 (0.003) | 0.390 (0.004) | 0.389 (0.006) | 0.386 (0.004) | 0.502 (0.013) | 0.505 (0.011) | 0.503 (0.008) | 0.495 (0.004) |
| LR+sLDA | 0.416 (0.007) | 0.426 (0.003) | 0.430 (0.004) | 0.432 (0.002) | 0.533 (0.007) | 0.564 (0.003) | 0.567 (0.006) | 0.571 (0.002) |
| LR+BPsLDA | 0.394 (0.004) | 0.396 (0.005) | 0.400 (0.005) | 0.419 (0.009) | **0.593 (0.003)** | *0.597* (0.002) | *0.597* (0.002) | *0.603* (0.002) |
| rSCHOLAR | **0.494** (0.005) | **0.495** (0.003) | **0.495** (0.003) | **0.494** (0.004) | 0.520 (0.02) | 0.548 (0.02) | 0.563 (0.01) | 0.571 (0.01) |
| **BTR** | *0.439* (0.008) | *0.447* (0.005) | *0.453* (0.003) | *0.454* (0.003) | *0.586* (0.007) | **0.615** (0.006) | **0.627** (0.004) | **0.630** (0.001) |
| | | | | Perplexity (lower is better) | | | | |
| LDA+LR | 538 (3) | 498 (2) | 476 (2) | 454 (1) | 1544 (5) | 1447 (4) | 1388 (4) | 1306 (4) |
| GSM+LR | **371 (6)** | **359 (11)** | **356 (14)** | **369 (8)** | **1500 (52)** | *1444* (29) | 1463 (21) | 1431 (34) |
| LR+sLDA | *535* (2) | 491 (1) | *463* (1) | *436* (2) | 1544 (6) | *1444* (6) | 1382 (5) | *1294* (5) |
| rSCHOLAR | 941 (134) | 1429 (163) | 2110 (396) | 5014 (1314) | 1744 (158) | 1918 (138) | 2216 (164) | 2814 (383) |
| **BTR** | *535* (2) | *490* (1) | *463* (2) | 437 (1) | *1540* (5) | **1443 (4)** | **1379 (4)** | **1291 (5)** |

Average over 20 runs per model. Standard deviation in brackets.Best model in **bold**. Second best model in *italics*.

We also tested sLDA+LR and a pure sLDA, which performed consistently worse so they are not included for the sake of brevity. For example, for $K = 50$, sLDA+LR achieved $pR^2$ of 0.420 and 0.564 for Booking and Yelp respectively, compared to 0.432 and 0.571 for LR+sLDA. Standalone sLDA achieves 0.356 and 0.526 respectively.

### B.6.2 Model parametrisations

This section provides an overview over all used and tested hyperparameter settings across all models in our benchmark list. Table B.5 lists all hyperparameter settings pertaining to topic model components. Table B.6 provides an overview over all used neural network hyperparameters. B.7 summarises the iteration and stopping criteria for all models.

Table B.5: Topic model hyperparameters

| | K | $\alpha$ | $\eta$ | $\mu_{\text{ntm}}$ | $\sigma_{\text{ntm}}$ | $a_0$ | $b_0$ | $m_0$ | $S_0$ |
|---|---|---|---|---|---|---|---|---|---|
| LDA | [10,20,30,50] | [0.1,0.5,1] | [0.001,0.01,0.1] | - | - | - | - | - | - |
| sLDA | [10,20,30,50] | [0.1,0.5,1] | [0.001,0.01,0.1] | - | - | - | - | - | - |
| BPsLDA | [10,20,30,50] | [0.1,0.5,1] | [0.001,0.01,0.1] | - | - | - | - | - | - |
| BTR | [10,20,30,50,100] | [0.1,**0.5**,1] | [0.001,**0.01**,0.1] | - | - | [0,1.5,**3**,**4**] | [0,**2**,4] | 0 | 2 |
| GSM | [10,20,30,50] | - | - | 0 | 1 | - | - | - | - |
| TAM | 100 | - | - | 0 | 1 | - | - | - | - |

*Bold parameter specifications were used for reported results in paper, unless stated otherwise. For Booking default $a_0 = 3$, for Yelp $a_0 = 4$.*

Table B.6: Neural network hyperparameters

| | HidLaySize $\theta$ | BatchSize | LearnRate | DropOut KeepRate | EmbedSize | HidLaySize RNN | TAM-thresh | nHidLayers BPsLDA |
|---|---|---|---|---|---|---|---|---|
| GSM | 64 | 64 | 1.00E-03 | [0.5,0.8,1]* | - | - | - | - |
| TAM | 64 | 64 | 1.00E-03 | 0.8 | 100 | 64 | 1/K | - |
| aRNN | - | 64 | 1.00E-03 | 0.8 | 100 | 64 | - | - |
| BPsLDA | - | 1050 | 1.00E-02 | - | - | - | - | 10 |

** best results (which occurred under no dropout) were reported in benchmarks*

Table B.7: Iteration parameters

| | E-step iters | M-step iters | max. EM-iters | burn-in | max. epochs | Gibbs iters (thinning) |
|---|---|---|---|---|---|---|
| LDA | - | - | 50*** | 100 | - | 1000 (5) |
| sLDA | [100,250,500]** | 2500 | 50*** | 20 | - | - |
| BTR | [100,250,500]** | 2500 | 50*** | 20 | - | - |
| GSM | - | - | - | - | 100*** | - |
| TAM | - | - | - | - | 100*** | - |
| BPsLDA | - | - | - | - | 50*** | - |

*** no noticeable performance difference observed, therefore all results reported based on 100 E-step.*
**** best model achieved substantially before max. iterations reached.*

**Further notes on benchmark model specifications:**

**For TAM and aRNN**, the sequence length in the RNN component (ie. the maximum number of words per document) is 305 for Booking and 572 for Yelp which corresponds to the longest review in each respective data set. We therefore work with the full text of each review.

**BPsLDA** changes its behaviour quite drastically when $\alpha$ is set in an area $1 \leq \alpha \leq 2$, where it strongly increases its predictive performance ($pR^2$) at the cost of its document modelling performance (perplexity). This can be seen in the original paper Chen et al. (2015). We included $\alpha = 1$ in the robustness test range and BTR is still generally on par with BPsLDA in this specific case for low $K$ and does better for $K > 30$. Even when including $\alpha = 1$ in the robustness test range, BTR still outperforms BPsLDA and all other models across all hyperparameter settings, except $K = 10$ in the Yelp dataset, where BTR is a close second.

### B.6.3 Robustness Tests

Robustness test across all topic models with LDA-like structure and Dirichlet hyperparameters for document-topic and word-topic distributions.

We assess the robustness of our findings to changes in the Dirichlet hyperparameters $\alpha$ and $\eta$. These hyperparameters act as priors on the topic-document distributions ($\boldsymbol{\beta}$) and word-topic distributions ($\boldsymbol{\theta}$), respectively. Table B.8 shows the results.

In terms of $pR^2$, BTR continues to perform best for all settings. We generally find that the BTR prediction performance is robust to hyperparameter changes. Evaluating the perplexity scores, we see more fluctuation across all models, which is unsurprising since those hyperparameter directly affect the generative topic modelling processes. BTR remains on par with its sLDA counterpart.

Table B.8: Sensitivity to hyperparameters $\alpha$ and $\beta$ (K = 20)

| | | $\alpha$ | | | $\eta$ | | |
|---|---|---|---|---|---|---|---|
| *Metric* | *Model* | 0.1 | 0.5 | 1 | 0.001 | 0.01 | 0.1 |
| 4*Yelp $pR^2$ | LR-LDA | 0.473 | 0.530 | 0.550 | 0.316 | 0.530 | 0.521 |
| | LR-sLDA | 0.558 | 0.564 | 0.559 | 0.562 | 0.564 | 0.568 |
| | LR-BPsLDA | 0.602 | 0.597 | 0.608 | 0.607 | 0.597 | 0.608 |
| | BTR | **0.611** | **0.615** | **0.613** | **0.611** | **0.615** | **0.624** |
| 3*Yelp perplexity | LR-LDA | 1511 | 1448 | 1445 | 1472 | 1447 | **1470** |
| | LR-sLDA | 1497 | 1444 | **1431** | **1441** | 1444 | 1491 |
| | BTR | **1490** | **1443** | 1441 | 1456 | **1443** | 1478 |
| 4*Booking $pR^2$ | LR-LDA | 0.397 | 0.410 | 0.409 | 0.405 | 0.410 | 0.406 |
| | LR-sLDA | 0.430 | 0.426 | 0.432 | 0.422 | 0.426 | 0.433 |
| | LR-BPsLDA | 0.409 | 0.396 | **0.453** | 0.395 | 0.396 | 0.393 |
| | BTR | **0.451** | **0.447** | 0.452 | **0.443** | **0.447** | **0.455** |
| 3*Booking perplexity | LR-LDA | 515 | 498 | 514 | 505 | 498 | **512** |
| | LR-sLDA | **502** | **491** | 504 | **484** | **491** | 516 |
| | BTR | 503 | **491** | **503** | 489 | **491** | 515 |

### B.6.3.1 Further Robustness Tests - Booking

Table B.9 provides an extended robustness test on the predictive performance of the benchmark topic models across hyperparameters. BTR continues to be the best performing model throughout. Table B.10 summarises robustness tests in terms of perplexity scores. BTR achieves almost identical perplexity scores as sLDA whilst achieving higher $pR^2$ throughout.

Table B.9: Booking - $pR^2$ for different hyperparameter settings across topic benchmark models, best model in bold.

| (K=10) | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LR-LDA | 0.378 | 0.4 | 0.408 | 0.397 | 0.4 | 0.398 | | | | |
| LR-sLDA | 0.42 | 0.416 | 0.403 | 0.401 | 0.416 | 0.422 | | | | |
| LR-BPsLDA | 0.396 | 0.394 | 0.439 | 0.393 | 0.394 | 0.396 | | | | |
| **BTR** | 0.446 | 0.439 | 0.435 | 0.418 | 0.439 | **0.452** | 0.439 | 0.435 | 0.437 | 0.446 |
| (K=20) | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
| LR-LDA | 0.397 | 0.41 | 0.409 | 0.405 | 0.41 | 0.406 | | | | |
| LR-sLDA | 0.43 | 0.426 | 0.432 | 0.422 | 0.426 | 0.433 | | | | |
| LR-BPsLDA | 0.409 | 0.396 | 0.453 | 0.395 | 0.396 | 0.393 | | | | |
| **BTR** | 0.451 | 0.447 | 0.452 | 0.443 | 0.447 | **0.455** | 0.447 | 0.45 | 0.45 | 0.443 |
| (K=30) | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
| LR-LDA | 0.399 | 0.417 | 0.423 | 0.417 | 0.417 | 0.413 | | | | |
| LR-sLDA | 0.434 | 0.43 | 0.428 | 0.417 | 0.43 | 0.427 | | | | |
| LR-BPsLDA | 0.424 | 0.4 | 0.451 | 0.401 | 0.4 | 0.402 | | | | |
| **BTR** | 0.455 | 0.453 | 0.455 | 0.444 | 0.453 | **0.459** | 0.453 | 0.453 | 0.447 | 0.449 |
| (K=50) | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
| LR-LDA | 0.415 | 0.426 | 0.428 | 0.418 | 0.426 | 0.420 | | | | |
| LR-sLDA | 0.434 | 0.432 | 0.43 | 0.429 | 0.432 | 0.436 | | | | |
| **LR-BPsLDA** | **0.461** | 0.419 | 0.449 | 0.411 | 0.419 | 0.418 | | | | |
| **BTR** | **0.461** | 0.454 | 0.459 | 0.446 | 0.454 | 0.459 | 0.454 | 0.455 | 0.452 | 0.451 |

*Default model was $\alpha = 0.5$, $\eta = 0.01$, $a_0 = 3$, $b_0 = 2$.*
*Robustness tests kept all hyperparameters at default, then changing one hyperparameter at a time.*

Table B.10: Booking - perplexity scores for different hyperparameter settings across topic benchmark models, best model in bold.

| K=10 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 562 | 538 | 539 | 539 | 538 | 545 | | | | |
| LR-sLDA | 557 | 535 | 539 | 534 | 535 | 554 | | | | |
| **BTR** | 556 | 535 | 538 | **528** | 535 | 548 | 535 | 535 | 537 | 536 |

| K=20 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 515 | 498 | 514 | 505 | 498 | 512 | | | | |
| **LR-sLDA** | 502 | 491 | 504 | **484** | 491 | 516 | | | | |
| BTR | 503 | 490 | 503 | 489 | 490 | 515 | 490 | 491 | 490 | 491 |

| K=30 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 479 | 476 | 502 | 484 | 476 | 492 | | | | |
| **LR-sLDA** | 471 | 463 | 486 | **454** | 463 | 499 | | | | |
| BTR | 470 | 463 | 483 | 457 | 463 | 500 | 463 | 463 | 463 | 463 |

| K=50 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (3,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 442 | 454 | 494 | 460 | 454 | 476 | | | | |
| **LR-sLDA** | 431 | 436 | 466 | **421** | 436 | 492 | | | | |
| BTR | 430 | 437 | 467 | 423 | 437 | 492 | 437 | 436 | 439 | 437 |

*Default model was $\alpha = 0.5$, $\eta = 0.01$, $a_0 = 3$, $b_0 = 2$.*
*Robustness tests kept all hyperparameters at default, then changing one hyperparameter at a time.*

#### B.6.3.2 Further Robustness Tests - Yelp

Table B.11 provides an extended robustness test on the predictive performance of the benchmark topic models across hyperparameters. BTR continues to be the best performing model throughout, apart from the K=10 case, where it is a close second. Table B.12 summarises robustness tests in terms of perplexity scores. BTR achieves almost identical perplexity scores as sLDA whilst achieving higher $pR^2$ throughout.

Table B.11: Yelp - $pR^2$ for different hyperparameter settings across topic benchmark models, best model in bold.

| K=10 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LR-LDA | 0.476 | 0.498 | 0.515 | 0.503 | 0.498 | 0.49 | | | | |
| LR-sLDA | 0.523 | 0.533 | 0.539 | 0.52 | 0.533 | 0.527 | | | | |
| **LR-BPsLDA** | 0.596 | 0.593 | **0.606** | 0.595 | 0.593 | 0.592 | | | | |
| BTR | 0.592 | 0.586 | 0.593 | 0.575 | 0.586 | 0.596 | 0.586 | 0.588 | 0.578 | 0.59 |

| K=20 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LR-LDA | 0.473 | 0.53 | 0.55 | 0.483 | 0.53 | 0.521 | | | | |
| LR-sLDA | 0.558 | 0.564 | 0.559 | 0.562 | 0.564 | 0.568 | | | | |
| LR-BPsLDA | 0.602 | 0.597 | 0.608 | 0.607 | 0.597 | 0.608 | | | | |
| **BTR** | 0.611 | 0.615 | 0.613 | 0.611 | 0.615 | **0.624** | 0.615 | 0.62 | 0.593 | 0.621 |

| K=30 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LR-LDA | 0.499 | 0.561 | 0.563 | 0.547 | 0.561 | 0.565 | | | | |
| LR-sLDA | 0.565 | 0.567 | 0.563 | 0.567 | 0.567 | 0.56 | | | | |
| LR-BPsLDA | 0.609 | 0.597 | 0.607 | 0.599 | 0.597 | 0.599 | | | | |
| **BTR** | 0.624 | **0.627** | 0.612 | 0.608 | **0.627** | 0.622 | **0.627** | 0.623 | **0.627** | 0.626 |

| K=50 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LR-LDA | 0.523 | 0.586 | 0.591 | 0.571 | 0.586 | 0.582 | | | | |
| LR-sLDA | 0.573 | 0.571 | 0.564 | 0.556 | 0.571 | 0.573 | | | | |
| LR-BPsLDA | 0.612 | 0.603 | 0.606 | 0.604 | 0.603 | 0.604 | | | | |
| **BTR** | **0.632** | 0.630 | 0.623 | 0.621 | 0.630 | **0.632** | 0.630 | 0.629 | 0.629 | 0.628 |

Table B.12: Yelp - perplexity scores for different hyperparameter settings across topic benchmark models, best model in bold.

| K=10 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 1586 | 1544 | 1532 | 1557 | 1544 | 1552 | 1544 | | | |
| **LR-sLDA** | 1583 | 1544 | **1530** | 1561 | 1544 | 1554 | 1544 | | | |
| BTR | 1588 | 1540 | 1534 | 1565 | 1540 | 1546 | 1540 | 1539 | 1548 | 1547 |

| K=20 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 1511 | 1447 | 1445 | 1472 | 1447 | 1469 | 1447 | | | |
| **LR-sLDA** | 1497 | 1444 | **1431** | 1441 | 1444 | 1491 | 1444 | | | |
| BTR | 1490 | 1443 | 1441 | 1456 | 1443 | 1478 | 1443 | 1443 | 1445 | 1441 |

| K=30 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | $\eta = 0.1$ | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 1434 | 1388 | 1390 | 1412 | 1388 | 1415 | 1388 | | | |
| LR-sLDA | 1436 | 1382 | 1383 | 1395 | 1382 | 1442 | 1382 | | | |
| **BTR** | 1434 | **1379** | 1385 | 1390 | **1379** | 1448 | **1379** | 1378 | 1389 | 1379 |

| K=50 | $\alpha = 0.1$ | $\alpha = 0.5$ | $\alpha = 1$ | $\eta = 0.001$ | $\eta = 0.01$ | eta = 0.1 | a,b = (4,2) | a,b = (0,0) | a,b = (3,4) | a,b = (1.5,4) |
|---|---|---|---|---|---|---|---|---|---|---|
| LDA | 1352 | 1306 | 1325 | 1334 | 1306 | 1356 | 1306 | | | |
| LR-sLDA | 1349 | 1294 | 1310 | 1309 | 1294 | 1404 | 1294 | | | |
| **BTR** | 1338 | **1291** | 1303 | 1288 | **1291** | 1405 | **1291** | 1293 | 1294 | 1292 |

### B.6.4 Estimated Topics

The below tables are an extended version of the corresponding table in the paper. The show the top 3 negative and positive topics for $K = [10, 30, 100]$. Inspecting the top words in each of these topics compared with its regression coefficient, BTR models highly interpretable topics -

at least as interpretable as LDA or sLDA. At the same time BTR achieves substantially better prediction performances throughout all model specifications (see previous section).

Table B.13: Top 3 positive and negative topics for *Yelp* (K = 10)

| | Pos1 | Pos2 | Pos3 | Neg3 | Neg2 | Neg1 |
|---|---|---|---|---|---|---|
| 6*BTR topics | food | restaur | time | store | food | order |
| | great | dish | hair | locat | chicken | us |
| | place | lobster | back | like | good | ask |
| | servic | menu | work | can | order | servic |
| | friend | food | will | price | rice | wait |
| | love | order | day | go | dish | food |
| BTR regr. weights | 4.3 | 1.7 | 1.5 | -0.1 | -0.5 | -8.8 |
| 6*sLDA topics | food | time | locat | coffe | us | like |
| | place | hair | store | tea | order | place |
| | great | back | can | tri | ask | go |
| | servic | work | find | place | servic | much |
| | good | will | place | ice | tabl | im |
| | time | day | staff | cream | time | realli |
| sLDA regr. weights | 2.7 | 1.7 | 1.2 | 0.1 | -3.7 | -4.5 |
| 6*LDA topics | food | place | place | store | fri | order |
| | great | coffe | great | like | burger | us |
| | servic | good | good | locat | order | food |
| | restaur | tri | friend | can | like | servic |
| | dish | tea | bar | find | good | time |
| | menu | great | drink | go | chees | ask |
| LDA regr. weights | 1.2 | 0.7 | 0.5 | -0.1 | -0.4 | -2.6 |

Table B.14: Top 3 positive and negative topics for *Yelp* (K = 30)

| | Pos1 | Pos2 | Pos3 | Neg3 | Neg2 | Neg1 |
|---|---|---|---|---|---|---|
| 6*BTR topics | best | great | restaur | us | ask | like |
| | plac | friend | menu | order | said | disappoint |
| | alway | servic | dish | tabl | custom | better |
| | love | staff | wine | food | told | tast |
| | ever | recommend | steak | server | never | noth |
| | toronto | amaz | perfect | came | say | bad |
| BTR regr. weights | 6.9 | 6.0 | 2.1 | -3.9 | -8.4 | -13.3 |
| 6*sLDA topics | great | time | im | seem | ask | like |
| | love | alway | review | like | never | food |
| | amaz | go | place | much | custom | good |
| | recommend | year | star | make | said | place |
| | servic | never | go | think | servic | tast |
| | friend | everi | give | thing | told | better |
| sLDA regr. weights | 3.7 | 3.1 | 3.1 | -2.3 | -6.4 | -7.1 |
| 6*LDA topics | great | toronto | restaur | us | ask | like |
| | friend | visit | menu | tabl | custom | tast |
| | love | make | dish | order | said | disappoint |
| | amaz | love | wine | food | servic | better |
| | place | made | dessert | came | told | bad |
| | servic | best | dinner | server | manag | noth |
| LDA regr. weights | 3.0 | 1.6 | 1.3 | -1.2 | -4.9 | -8.3 |

Table B.15: Top 3 positive and negative topics for *Yelp* (K = 100)

| | Pos1 | Pos2 | Pos3 | Neg3 | Neg2 | Neg1 |
|---|---|---|---|---|---|---|
| 6*BTR topics | love | definit | best | custom | never | disappoint |
| | delici | ever | amaz | ask | worst | tast |
| | definit | toronto | everi | said | ever | bland |
| | perfect | citi | friend | manag | money | dri |
| | tri | far | free | rude | bad | better |
| | super | amaz | alway | servic | terribl | lack |
| BTR regr. weights | 5.9 | 5.2 | 5.0 | -8.4 | -12.5 | -14.5 |
| 6*sLDA topics | alway | will | amaz | ask | tast | disappoint |
| | time | definit | definit | said | like | bad |
| | usual | servic | love | told | felt | cold |
| | come | friend | great | back | disappoint | worst |
| | never | return | place | went | better | dri |
| | everi | back | everyth | want | wasnt | lack |
| sLDA regr. weights | 4.1 | 4.0 | 3.9 | -7.0 | -8.0 | -11.3 |
| 6*LDA topics | love | best | experi | money | never | tast |
| | amaz | toronto | made | go | bad | like |
| | delici | ever | make | will | ever | disappoint |
| | place | citi | feel | never | worst | meat |
| | absolut | far | first | pay | terribl | bland |
| | super | visit | felt | spend | experi | dri |
| LDA regr. weights | 5.4 | 4.6 | 3.8 | -5.9 | -10.8 | -10.9 |

Table B.16: Top 3 positive and negative topics for *Booking* (K = 10)

| | Pos1 | Pos2 | Pos3 | Neg3 | Neg2 | Neg1 |
|---|---|---|---|---|---|---|
| 6*BTR topics | hotel | room | room | room | check | hotel |
| | stay | locat | great | bed | book | room |
| | staff | staff | love | shower | room | locat |
| | would | good | hotel | bathroom | us | small |
| | help | clean | view | small | hotel | good |
| | everyth | comfort | bar | clean | arriv | price |
| BTR regr. weights | 2.8 | 1.7 | 1.5 | -1.0 | -1.1 | -5.7 |
| 6*sLDA topics | hotel | room | room | room | room | room |
| | stay | good | great | bed | night | hotel |
| | staff | locat | love | bathroom | window | locat |
| | would | staff | hotel | shower | work | small |
| | help | clean | view | small | floor | staff |
| | like | breakfast | nice | comfort | air | posit |
| sLDA regr. weights | 2.4 | 1.3 | 1.3 | -0.4 | -0.6 | -5.6 |
| 6*LDA topics | hotel | hotel | neg | check | room | room |
| | stay | great | staff | room | shower | hotel |
| | staff | love | locat | book | bathroom | good |
| | help | room | friendli | hotel | work | locat |
| | would | view | great | us | bed | breakfast |
| | noth | locat | help | time | air | price |
| LDA regr. weights | 1.3 | 1.2 | 1.0 | -1.3 | -1.4 | -2.0 |

Table B.17: Top 3 positive and negative topics for *Booking* (K = 30)

| | Pos1 | Pos2 | Pos3 | Neg3 | Neg2 | Neg1 |
|---|---|---|---|---|---|---|
| 6*BTR topics | us | stay | room | room | ask | room |
| | staff | would | locat | small | us | hotel |
| | made | hotel | great | bed | day | old |
| | upgrad | staff | staff | size | recept | poor |
| | stay | love | bit | locat | call | star |
| | welcom | recommend | littl | bathroom | back | bad |
| BTR regr. weights | 2.7 | 2.5 | 2.3 | -2.5 | -3.0 | -9.0 |
| 6*sLDA topics | staff | hotel | us | book | room | hotel |
| | friendli | love | upgrad | charg | need | room |
| | great | beauti | staff | hotel | locat | bad |
| | help | decor | room | pay | old | star |
| | locat | modern | stay | check | look | poor |
| | neg | great | love | day | smell | posit |
| sLDA regr. weights | 2.1 | 1.8 | 1.7 | -1.4 | -2.1 | -9.7 |
| 6*LDA topics | stay | stay | hotel | us | room | hotel |
| | hotel | hotel | love | ask | locat | like |
| | made | would | beauti | one | good | star |
| | like | recommend | great | recept | need | realli |
| | feel | definit | decor | day | old | much |
| | realli | love | staff | call | valu | best |
| LDA regr. weights | 2.4 | 2.1 | 1.9 | -2.5 | -2.7 | -3.4 |

Table B.18: Top 3 positive and negative topics for *Booking* (K = 100)

| | Pos1 | Pos2 | Pos3 | Neg3 | Neg2 | Neg1 |
|---|---|---|---|---|---|---|
| 6*BTR topics | staff | hotel | love | old | room | poor |
| | help | wonder | great | look | small | posit |
| | friendli | beauti | staff | carpet | tini | servic |
| | excel | love | littl | tire | bathroom | bad |
| | especi | experi | fab | furnitur | noisi | never |
| | wonder | fabul | especi | need | far | rude |
| BTR regr. weights | 4.3 | 4.1 | 3.3 | -6.1 | -7.3 | -14.2 |
| 6*sLDA topics | love | room | great | old | hotel | bad |
| | beauti | small | locat | dirti | star | poor |
| | amaz | posit | neg | bathroom | expect | recept |
| | fantast | size | perfect | carpet | rate | posit |
| | fabul | bit | awesom | wall | thi | even |
| | wonder | expect | super | look | basic | never |
| sLDA regr. weights | 3.7 | 3.6 | 2.8 | -5.8 | -6.0 | -14.3 |
| 6*LDA topics | love | great | bit | hotel | old | recept |
| | amaz | locat | littl | star | dirti | manag |
| | everyth | neg | nice | rate | carpet | rude |
| | noth | staff | locat | expect | look | receptionist |
| | perfect | awesom | breakfast | disappoint | wall | check |
| | absolut | perfect | good | thi | furnitur | guest |
| LDA regr. weights | 3.6 | 3.1 | 2.8 | -5.3 | -6.6 | -7.6 |

### B.6.5 Computation Times

Table B.19 shows the time taken for 100 E-step iterations on a single 2.8GHz processor on the Booking data and 300-400 seconds on the Yelp data. We found that 100 E-step iterations is typically sufficient for the best performance and the model typically converges after between 10-25 EM iterations. A typical 30 topic model on Yelp data thus took around 1 hour to converge, and around 20 minutes for Booking. Computation time scales roughly linearly in the number of topics and total number of words across all documents. This is because the evaluation of the $K$-dimensional multinomial distribution for each $z_{d,n}$ (equation (B.16)) is the principle computational challenge.

Table B.19: Computational time

| Dataset | K | 100 E-step iters |
|---|---|---|
| 5*Yelp | 10 | 50s |
| | 20 | 110s |
| | 30 | 200s |
| | 50 | 320s |
| | 100 | 740s |
| 5*Booking | 10 | 18s |
| | 20 | 33s |
| | 30 | 50s |
| | 50 | 79s |
| | 100 | 200s |

*Note*: Yelp data has roughly 3 times as many words as Booking.com data

# C

# Appendix: Central Bank Speeches and High-Frequency Market Responses

## C.1 List of Relevant Greenbook Sections

| **GDP** | **CPI** | **Unemployment** |
|---|---|---|
| Ec.GDP | Ec.Prices | Ec.Labor |
| For.Ec.Overview | For.CostPrice | For.Labor |
| For.Ec.Summary | Ec.Wages | |
| For.Outlook | | |
| For.HH | | |
| For.G | | |
| For.Inven | | |
| For.BusInvest | | |
| For.Trade | | |

Table C.1: Considered Greenbook sections per economic indicator. EC = Economic Conditions Section, For = Forecasts Section

## C.2 Lists of Stocks and Bonds

Table C.2: Stock tickers and names

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| AAPL | Apple | AXP | American | BA | Boeing | CAT | Caterpillar |
| CSCO | Cisco | CVX | Chevron | DIS | Disney | HD | Home |
| IBM | IBM | INTC | Intel | JNJ | Johnson | KO | Coca-Cola |
| MCD | McDonald's | MMM | 3M | MRK | Merck | MSFT | MSFT |
| NKE | Nike | PFE | Pfizer | UNH | UnitedHealth | VZ | Verizon |
| WMT | Wal-Mart | XOM | Exxon | | | | |

Table C.3: Bond names and maturities

| | | | |
|---|---|---|---|
| US Treasury Note Futures: | 2-Year | 5-Year | 10-Year |

## C.3 Detailed Results - Language to Forecast Mapping

Table C.4: CPI mapping and fit performance

| Model \Predictive $R^2$ | score test | score val | score train | data source |
|---|---|---|---|---|
| **MM Neural Topic Model (non-lin)** | **0.735** | **0.830** | 0.670 | joint MM tabular + topics |
| **MM Neural Topic Model (linear)** | **0.640** | **0.650** | 0.600 | joint MM tabular + topics |
| ExtraTreesMSE_BAG_L1 | 0.588 | 0.084 | 0.880 | tabular |
| RandomForestMSE_BAG_L1 | 0.584 | 0.052 | 0.622 | tabular + topics |
| ExtraTreesMSE_BAG_L1 | 0.584 | 0.089 | 0.595 | tabular + topics |
| RandomForestMSE_BAG_L1 | 0.568 | 0.047 | 0.876 | tabular |
| KNeighborsUnif_BAG_L1 | 0.559 | 0.141 | 0.460 | tabular + topics |
| KNeighborsDist_BAG_L1 | 0.549 | 0.128 | 0.798 | tabular + topics |
| KNeighborsUnif_BAG_L1 | 0.520 | 0.152 | 0.439 | tabular + tfidf |
| KNeighborsDist_BAG_L1 | 0.519 | 0.146 | 1.000 | tabular + tfidf |
| KNeighborsUnif_BAG_L1 | 0.516 | 0.142 | 0.442 | tabular |
| NeuralNetFastAI_BAG_L1 | 0.515 | 0.233 | 0.251 | tabular + topics |
| KNeighborsDist_BAG_L1 | 0.513 | 0.121 | 1.000 | tabular |
| OLS | 0.512 | | 0.288 | tabular |
| NeuralNetFastAI_BAG_L1 | 0.494 | 0.272 | 0.594 | tabular |
| RandomForestMSE_BAG_L1 | 0.482 | 0.103 | 0.883 | tabular + tfidf |
| WeightedEnsemble_L2 | 0.475 | 0.302 | 0.565 | tabular |
| CatBoost_BAG_L1 | 0.386 | 0.200 | 0.698 | tabular |
| CatBoost_BAG_L1 | 0.384 | 0.170 | 0.905 | tabular + tfidf |
| XGBoost_BAG_L1 | 0.377 | 0.169 | 0.595 | tabular + topics |
| XGBoost_BAG_L1 | 0.374 | 0.155 | 0.937 | tabular + tfidf |
| LightGBMXT_BAG_L1 | 0.373 | 0.126 | 0.295 | tabular |
| XGBoost_BAG_L1 | 0.368 | 0.152 | 0.770 | tabular |
| WeightedEnsemble_L2 | 0.358 | 0.284 | 0.370 | tabular + topics |
| LightGBMLarge_BAG_L1 | 0.357 | 0.080 | 0.646 | tabular + tfidf |
| LightGBM_BAG_L1 | 0.327 | 0.136 | 0.294 | tabular |
| WeightedEnsemble_L2 | 0.299 | 0.305 | 0.953 | tabular + tfidf |
| LightGBM_BAG_L1 | 0.289 | 0.138 | 0.245 | tabular + topics |
| NeuralNetTorch_BAG_L1 | 0.269 | 0.210 | 0.128 | tabular + topics |
| NeuralNetTorch_BAG_L1 | 0.262 | 0.247 | 0.401 | tabular |
| XGBoost_BAG_L1 | 0.260 | 0.056 | 0.783 | tabular + embeddings |
| LightGBMXT_BAG_L1 | 0.252 | 0.092 | 0.348 | tabular + tfidf |
| LightGBM_BAG_L1 | 0.252 | 0.131 | 0.368 | tabular + tfidf |
| LightGBMLarge_BAG_L1 | 0.251 | 0.139 | 0.302 | tabular |
| LightGBMLarge_BAG_L1 | 0.202 | 0.156 | 0.323 | tabular + topics |
| ExtraTreesMSE_BAG_L1 | 0.193 | 0.143 | 0.889 | tabular + tfidf |
| LightGBMLarge_BAG_L1 | 0.191 | 0.074 | 0.440 | tabular + embeddings |
| CatBoost_BAG_L1 | 0.177 | 0.250 | 0.525 | tabular + topics |
| LightGBMXT_BAG_L1 | 0.162 | 0.140 | 0.192 | tabular + topics |
| NeuralNetFastAI_BAG_L1 | 0.148 | 0.280 | 0.912 | tabular + tfidf |
| WeightedEnsemble_L2 | 0.132 | 0.139 | 0.573 | tabular + embeddings |
| CatBoost_BAG_L1 | 0.126 | 0.116 | 0.633 | tabular + embeddings |
| LightGBMXT_BAG_L1 | 0.116 | 0.001 | 0.520 | tabular + embeddings |
| LightGBM_BAG_L1 | 0.112 | -0.018 | 0.338 | tabular + embeddings |
| NeuralNetTorch_BAG_L1 | 0.095 | 0.153 | 0.500 | tabular + tfidf |
| NeuralNetTorch_BAG_L1 | -0.030 | 0.076 | 0.161 | tabular + embeddings |
| AutoGluon Multimodal Transformer | -0.292 | | -0.155 | multimodal embeddings |

Table C.5: GDP mapping and fit performance

| **Model \Predictive $R^2$** | **score test** | **score val** | **score train** | **data source** |
|---|---|---|---|---|
| **MM Neural Topic Model (lin)** | **0.825** | **0.426** | 0.372 | joint MM tabular + topics |
| **MM Neural Topic Model (non-lin)** | **0.797** | **0.371** | 0.483 | joint MM tabular + topics |
| WeightedEnsemble_L2 | 0.380 | 0.304 | 0.497 | tabular |
| OLS | 0.785 | 0.301 | | tabular |
| NeuralNetFastAI_BAG_L1 | 0.480 | 0.270 | 0.443 | tabular |
| WeightedEnsemble_L2 | 0.285 | 0.253 | 0.730 | tabular + topics |
| WeightedEnsemble_L2 | 0.268 | 0.240 | 0.752 | tabular + tfidf |
| WeightedEnsemble_L2 | 0.142 | 0.220 | 0.587 | tabular + embeddings |
| CatBoost_BAG_L1 | 0.249 | 0.211 | 0.552 | tabular |
| RandomForestMSE_BAG_L1 | 0.302 | 0.204 | 0.892 | tabular + tfidf |
| RandomForestMSE_BAG_L1 | 0.348 | 0.202 | 0.892 | tabular + topics |
| ExtraTreesMSE_BAG_L1 | 0.408 | 0.193 | 0.891 | tabular |
| ExtraTreesMSE_BAG_L1 | 0.381 | 0.192 | 0.890 | tabular + topics |
| ExtraTreesMSE_BAG_L1 | 0.111 | 0.188 | 0.891 | tabular + tfidf |
| CatBoost_BAG_L1 | 0.207 | 0.187 | 0.671 | tabular + tfidf |
| LightGBMXT_BAG_L1 | 0.203 | 0.178 | 0.322 | tabular |
| LightGBM_BAG_L1 | 0.154 | 0.172 | 0.367 | tabular |
| XGBoost_BAG_L1 | 0.141 | 0.171 | 0.580 | tabular + topics |
| CatBoost_BAG_L1 | 0.006 | 0.169 | 0.531 | tabular + topics |
| CatBoost_BAG_L1 | 0.101 | 0.169 | 0.552 | tabular + embeddings |
| LightGBM_BAG_L1 | 0.099 | 0.162 | 0.704 | tabular + embeddings |
| NeuralNetTorch_BAG_L1 | 0.461 | 0.160 | 0.341 | tabular |
| LightGBM_BAG_L1 | 0.101 | 0.159 | 0.734 | tabular + tfidf |
| KNeighborsUnif_BAG_L1 | 0.253 | 0.158 | 0.402 | tabular + tfidf |
| LightGBMLarge_BAG_L1 | 0.245 | 0.155 | 0.598 | tabular |
| KNeighborsDist_BAG_L1 | 0.256 | 0.151 | 1.000 | tabular + tfidf |
| NeuralNetTorch_BAG_L1 | 0.049 | 0.150 | 0.553 | tabular + tfidf |
| LightGBMXT_BAG_L1 | 0.120 | 0.150 | 0.348 | tabular + tfidf |
| RandomForestMSE_BAG_L1 | 0.394 | 0.150 | 0.885 | tabular |
| LightGBMLarge_BAG_L1 | 0.111 | 0.149 | 0.536 | tabular + topics |
| LightGBMLarge_BAG_L1 | 0.181 | 0.149 | 0.665 | tabular + embeddings |
| XGBoost_BAG_L1 | 0.119 | 0.142 | 0.567 | tabular |
| NeuralNetFastAI_BAG_L1 | 0.060 | 0.136 | 0.797 | tabular + tfidf |
| KNeighborsDist_BAG_L1 | 0.255 | 0.132 | 1.000 | tabular |
| KNeighborsUnif_BAG_L1 | 0.248 | 0.130 | 0.407 | tabular |
| LightGBM_BAG_L1 | 0.111 | 0.126 | 0.496 | tabular + topics |
| LightGBMXT_BAG_L1 | 0.105 | 0.125 | 0.505 | tabular + embeddings |
| NeuralNetTorch_BAG_L1 | -0.071 | 0.123 | 0.275 | tabular + embeddings |
| NeuralNetTorch_BAG_L1 | 0.151 | 0.108 | 0.497 | tabular + topics |
| XGBoost_BAG_L1 | -0.015 | 0.107 | 0.663 | tabular + embeddings |
| LightGBMLarge_BAG_L1 | 0.108 | 0.095 | 0.581 | tabular + tfidf |
| XGBoost_BAG_L1 | 0.041 | 0.083 | 0.564 | tabular + tfidf |
| KNeighborsUnif_BAG_L1 | 0.286 | 0.081 | 0.400 | tabular + topics |
| KNeighborsDist_BAG_L1 | 0.274 | 0.074 | 1.000 | tabular + topics |
| LightGBMXT_BAG_L1 | 0.097 | 0.049 | 0.318 | tabular + topics |
| TextPredictor_BAG_L1 | -0.077 | -0.123 | -0.103 | tabular + embeddings |
| NeuralNetFastAI_BAG_L1 | 0.407 | -0.126 | 0.438 | tabular + topics |
| AutoGluon Multimodal Transformer | -0.044 | | 0.013 | multimodal transformer |

Table C.6: Unemployment mapping and fit performance

| **Model \Predictive $R^2$** | **score_test** | **score_val** | **score_train** | **data source** |
|---|---|---|---|---|
| **MM Neural Topic Model** (non-lin) | **0.208** | **0.457** | 0.285 | joint MM tabular + topics |
| WeightedEnsemble_L2 | -0.044 | 0.145 | 0.415 | tabular + embeddings |
| NeuralNetTorch_BAG_L1 | -0.152 | 0.122 | 0.313 | tabular + embeddings |
| WeightedEnsemble_L2 | -0.045 | 0.113 | 0.577 | tabular + tfidf |
| MM Neural Topic Model (linear) | 0.066 | 0.109 | 0.197 | joint MM tabular + topics |
| CatBoost_BAG_L1 | -0.055 | 0.104 | 0.690 | tabular + tfidf |
| LightGBMXT_BAG_L1 | -0.068 | 0.074 | 0.336 | tabular + tfidf |
| NeuralNetTorch_BAG_L1 | -0.029 | 0.070 | 0.394 | tabular + tfidf |
| WeightedEnsemble_L2 | 0.131 | 0.058 | 0.191 | tabular |
| WeightedEnsemble_L2 | -0.010 | 0.053 | 0.278 | tabular + topics |
| NeuralNetFastAI_BAG_L1 | 0.124 | 0.047 | 0.237 | tabular |
| CatBoost_BAG_L1 | 0.021 | 0.041 | 0.411 | tabular + embeddings |
| NeuralNetTorch_BAG_L1 | 0.106 | 0.033 | 0.098 | tabular |
| LightGBM_BAG_L1 | 0.006 | 0.027 | 0.349 | tabular + embeddings |
| LightGBM_BAG_L1 | -0.035 | 0.025 | 0.316 | tabular + tfidf |
| CatBoost_BAG_L1 | -0.003 | 0.021 | 0.260 | tabular + topics |
| CatBoost_BAG_L1 | 0.019 | 0.010 | 0.095 | tabular |
| RandomForestMSE_BAG_L1 | -0.072 | 0.008 | 0.868 | tabular + tfidf |
| NeuralNetTorch_BAG_L1 | -0.004 | 0.006 | 0.022 | tabular + topics |
| XGBoost_BAG_L1 | -0.112 | 0.006 | 0.883 | tabular + tfidf |
| LightGBMLarge_BAG_L1 | -0.001 | 0.001 | 0.594 | tabular + embeddings |
| LightGBMLarge_BAG_L1 | 0.002 | -0.003 | 0.109 | tabular + topics |
| ExtraTreesMSE_BAG_L1 | -0.045 | -0.003 | 0.868 | tabular + tfidf |
| LightGBMXT_BAG_L1 | -0.001 | -0.005 | 0.084 | tabular |
| LightGBMXT_BAG_L1 | 0.000 | -0.006 | 0.009 | tabular + topics |
| LightGBM_BAG_L1 | 0.000 | -0.007 | 0.015 | tabular + topics |
| LightGBMXT_BAG_L1 | -0.005 | -0.024 | 0.292 | tabular + embeddings |
| XGBoost_BAG_L1 | -0.043 | -0.027 | 0.495 | tabular + topics |
| LightGBM_BAG_L1 | -0.002 | -0.028 | 0.170 | tabular |
| LightGBMLarge_BAG_L1 | 0.013 | -0.034 | 0.094 | tabular |
| NeuralNetFastAI_BAG_L1 | 0.002 | -0.036 | 0.565 | tabular + tfidf |
| XGBoost_BAG_L1 | -0.061 | -0.041 | 0.624 | tabular + embeddings |
| LightGBMLarge_BAG_L1 | -0.045 | -0.044 | 0.519 | tabular + tfidf |
| NeuralNetFastAI_BAG_L1 | -0.016 | -0.058 | 0.025 | tabular + topics |
| RandomForestMSE_BAG_L1 | -0.005 | -0.101 | 0.855 | tabular + topics |
| XGBoost_BAG_L1 | -0.048 | -0.126 | 0.277 | tabular |
| ExtraTreesMSE_BAG_L1 | 0.008 | -0.144 | 0.849 | tabular |
| ExtraTreesMSE_BAG_L1 | 0.049 | -0.163 | 0.848 | tabular + topics |
| KNeighborsUnif_BAG_L1 | -0.013 | -0.185 | 0.188 | tabular + tfidf |
| KNeighborsUnif_BAG_L1 | -0.004 | -0.187 | 0.186 | tabular |
| KNeighborsUnif_BAG_L1 | -0.048 | -0.187 | 0.195 | tabular + topics |
| TextPredictor_BAG_L1 | -0.067 | -0.190 | -0.070 | tabular + embeddings |
| KNeighborsDist_BAG_L1 | -0.003 | -0.191 | 1.000 | tabular + tfidf |
| RandomForestMSE_BAG_L1 | -0.034 | -0.192 | 0.842 | tabular |
| KNeighborsDist_BAG_L1 | -0.030 | -0.210 | 1.000 | tabular + topics |
| KNeighborsDist_BAG_L1 | 0.003 | -0.215 | 1.000 | tabular |
| OLS | -0.377 | | 0.231 | tabular |
| AutoGluon Multimodal Transformer | -1.177 | | -0.737 | multimodal transformer |

## C.4 Detailed Results - Equity Markets, CPI Regimes

### C.4.1 High CPI Regime

Table C.7: Association between news and market volatility, equity markets, high CPI regime

| Target variable: $RV_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| CPI news pos. \| | 0.2740 | 0.070 | 3.936 | 0.000 | 0.138 | 0.410 |
| \| CPI news neg. \| | 0.1437 | 0.052 | 2.780 | 0.005 | 0.042 | 0.245 |
| \| GDP news pos. \| | 0.0820 | 0.164 | 0.499 | 0.618 | -0.240 | 0.404 |
| \| GDP news neg. \| | 0.0118 | 0.087 | 0.136 | 0.892 | -0.159 | 0.183 |
| \| U news pos. \| | 9.1621 | 2.098 | 4.368 | 0.000 | 5.051 | 13.273 |
| \| U news neg. \| | 0.1683 | 0.076 | 2.215 | 0.027 | 0.019 | 0.317 |
| $R^2$: 0.917 | Adj. $R^2$: 0.901 | n. obs.: 36 | Heteroscedasticity robust standard errors | | | |

Table C.8: Association between news and tail risk, equity markets, high CPI regime

| Target variable: $TR_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 3.1074 | 2.184 | 1.423 | 0.155 | -1.173 | 7.388 |
| \| News CPI neg. \| | 2.7033 | 1.791 | 1.509 | 0.131 | -0.808 | 6.215 |
| \| News GDP pos. \| | -1.5404 | 4.349 | -0.354 | 0.723 | -10.064 | 6.983 |
| \| News GDP neg. \| | 0.8172 | 1.466 | 0.557 | 0.577 | -2.056 | 3.690 |
| \| News U pos. \| | 187.3136 | 13.601 | 13.772 | 0.000 | 160.657 | 213.970 |
| \| News U neg. \| | 2.3664 | 2.479 | 0.955 | 0.340 | -2.492 | 7.225 |
| $R^2$: 0.683 | Adj. $R^2$: 0.619 | n. obs.: 36 | Heteroscedasticity robust standard errors | | | |

### C.4.2 Low CPI Regime

Table C.9: Association between news and market volatility, equity markets, low CPI regime

| Target variable: $RV_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.1657 | 0.048 | 3.457 | 0.001 | 0.072 | 0.260 |
| \| News CPI neg. \| | 0.1305 | 0.064 | 2.046 | 0.041 | 0.005 | 0.256 |
| \| News GDP pos. \| | 0.4317 | 0.170 | 2.546 | 0.011 | 0.099 | 0.764 |
| \| News GDP neg. \| | 0.1279 | 0.158 | 0.812 | 0.417 | -0.181 | 0.437 |
| \| News U pos. \| | 0.1730 | 0.160 | 1.084 | 0.278 | -0.140 | 0.486 |
| \| News U neg. \| | 0.1008 | 0.029 | 3.459 | 0.001 | 0.044 | 0.158 |
| $R^2$: 0.774 | Adj. $R^2$: 0.748 | n. obs.: 59 | Heteroscedasticity robust standard errors | | | |

Table C.10: Association between news and tail risk, equity markets, low CPI regime

| Target variable: $TR_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 1.5924 | 1.068 | 1.491 | 0.136 | -0.500 | 3.685 |
| \| News CPI neg. \| | 1.2883 | 1.368 | 0.942 | 0.346 | -1.393 | 3.970 |
| \| News GDP pos. \| | 10.3541 | 3.365 | 3.077 | 0.002 | 3.759 | 16.949 |
| \| News GDP neg. \| | 4.6929 | 2.575 | 1.823 | 0.068 | -0.354 | 9.740 |
| \| News U pos. \| | 3.5833 | 3.297 | 1.087 | 0.277 | -2.880 | 10.046 |
| \| News U neg. \| | -0.2576 | 0.663 | -0.388 | 0.698 | -1.557 | 1.042 |
| $R^2$: 0.622 | Adj. $R^2$: 0.580 | n. obs.: 59 | Heteroscedasticity robust standard errors | | | |

### C.4.3 Normal CPI Regime

Table C.11: Association between news and market volatility, equity markets, normal CPI regime

| Target variable: $RV_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.1013 | 0.070 | 1.447 | 0.148 | -0.036 | 0.238 |
| \| News CPI neg. \| | 0.2412 | 0.161 | 1.494 | 0.135 | -0.075 | 0.558 |
| \| News GDP pos. \| | 0.2766 | 0.199 | 1.392 | 0.164 | -0.113 | 0.666 |
| \| News GDP neg. \| | 0.1507 | 0.243 | 0.620 | 0.536 | -0.326 | 0.627 |
| \| News U pos. \| | 0.7982 | 0.909 | 0.878 | 0.380 | -0.984 | 2.580 |
| \| News U neg. \| | 0.1983 | 0.059 | 3.369 | 0.001 | 0.083 | 0.314 |
| $R^2$: 0.771 | Adj. $R^2$: 0.749 | n. obs.: 70 | Heteroscedasticity robust standard errors | | | |

Table C.12: Association between news and tail risk, equity markets, normal CPI regime

| Target variable: $TR_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 1.5422 | 0.892 | 1.729 | 0.084 | -0.206 | 3.291 |
| \| News CPI neg. \| | 5.2256 | 3.888 | 1.344 | 0.179 | -2.395 | 12.847 |
| \| News GDP pos. \| | 2.6501 | 2.156 | 1.229 | 0.219 | -1.576 | 6.876 |
| \| News GDP neg. \| | -0.1986 | 2.686 | -0.074 | 0.941 | -5.463 | 5.066 |
| \| News U pos. \| | 4.6007 | 14.375 | 0.320 | 0.749 | -23.574 | 32.775 |
| \| News U neg. \| | 2.6626 | 0.624 | 4.264 | 0.000 | 1.439 | 3.886 |
| $R^2$: 0.593 | Adj. $R^2$: 0.555 | n. obs.: 70 | Heteroscedasticity robust standard errors | | | |

*: $p \leq 0.1$,**: $p \leq 0.05$,***: $p \leq 0.01$

## C.5 Detailed Results - Equity Markets, GDP Regimes

### C.5.1 High GDP Regimes

Table C.13: Association between news and market volatility, equity markets, high GDP regime

| Target variable: $RV_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.1078 | 0.068 | 1.586 | 0.113 | -0.025 | 0.241 |
| \| News CPI neg. \| | 0.0011 | 0.089 | 0.012 | 0.990 | -0.173 | 0.175 |
| \| News GDP pos. \| | 0.3347 | 0.292 | 1.148 | 0.251 | -0.237 | 0.906 |
| \| News GDP neg. \| | 0.1446 | 0.106 | 1.358 | 0.174 | -0.064 | 0.353 |
| \| News U pos. \| | 0.2226 | 0.216 | 1.032 | 0.302 | -0.200 | 0.645 |
| \| News U neg. \| | 0.2192 | 0.100 | 2.200 | 0.028 | 0.024 | 0.414 |
| $R^2$: 0.578 | Adj. $R^2$: 0.545 | n. obs.: 36 | Heteroscedasticity robust standard errors | | | |

Table C.14: Association between news and tail risk, equity markets, high GDP regime

| Target variable: $TR_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.9807 | 1.686 | 0.582 | 0.561 | -2.324 | 4.286 |
| \| News CPI neg. \| | 0.1379 | 1.242 | 0.111 | 0.912 | -2.297 | 2.573 |
| \| News GDP pos. \| | 2.0496 | 3.835 | 0.534 | 0.593 | -5.467 | 9.566 |
| \| News GDP neg. \| | 0.9372 | 2.394 | 0.391 | 0.695 | -3.756 | 5.630 |
| \| News U neg. \| | 4.2181 | 1.666 | 2.531 | 0.011 | 0.952 | 7.484 |
| $R^2$: 0.652 | Adj. $R^2$: 0.596 | n. obs.: 36 | Heteroscedasticity robust standard errors | | | |

### C.5.2 Low GDP Regime

Table C.15: Association between news and market volatility, equity markets, low GDP regime

| Target variable: $RV_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.2141 | 0.049 | 4.330 | 0.000 | 0.117 | 0.311 |
| \| News CPI neg. \| | 0.1031 | 0.035 | 2.980 | 0.003 | 0.035 | 0.171 |
| \| News GDP pos. \| | 0.5916 | 0.060 | 9.840 | 0.000 | 0.474 | 0.709 |
| \| News GDP neg. \| | 0.1953 | 0.071 | 2.767 | 0.006 | 0.057 | 0.334 |
| \| News U pos. \| | 0.5219 | 0.272 | 1.918 | 0.055 | -0.011 | 1.055 |
| \| News U neg. \| | 0.1513 | 0.026 | 5.795 | 0.000 | 0.100 | 0.202 |
| $R^2$: 0.796 | Adj. $R^2$: 0.778 | n. obs.: 44 | Heteroscedasticity robust standard errors | | | |

Table C.16: Association between news and tail risk, equity markets, low GDP regime

| Target variable: $TR_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | -0.4354 | 1.289 | -0.338 | 0.735 | -2.962 | 2.091 |
| \| News CPI neg. \| | 0.2284 | 1.619 | 0.141 | 0.888 | -2.944 | 3.401 |
| \| News GDP pos. \| | 7.3587 | 1.744 | 4.219 | 0.000 | 3.940 | 10.777 |
| \| News GDP neg. \| | 5.0831 | 2.958 | 1.718 | 0.086 | -0.715 | 10.881 |
| \| News U pos. \| | 8.7712 | 4.091 | 2.144 | 0.032 | 0.752 | 16.790 |
| \| News U neg. \| | 1.7789 | 0.894 | 1.991 | 0.047 | 0.028 | 3.530 |
| $R^2$: 0.565 | Adj. $R^2$: 0.517 | n. obs.: 44 | Heteroscedasticity robust standard errors | | | |

### C.5.3 Normal GDP Regime

Table C.17: Association between news and market volatility, equity markets, normal GDP regime

| Target variable: $RV_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.1482 | 0.095 | 1.560 | 0.119 | -0.038 | 0.334 |
| \| News CPI neg. \| | 0.1780 | 0.183 | 0.974 | 0.330 | -0.180 | 0.536 |
| \| News GDP pos. \| | 0.5693 | 0.335 | 1.700 | 0.089 | -0.087 | 1.226 |
| \| News GDP neg. \| | 0.3184 | 1.055 | 0.302 | 0.763 | -1.749 | 2.386 |
| \| News U pos. \| | 0.8327 | 0.593 | 1.405 | 0.160 | -0.329 | 1.994 |
| \| News U neg. \| | 0.1523 | 0.179 | 0.853 | 0.394 | -0.198 | 0.502 |
| $R^2$: 0.858 | Adj. $R^2$: 0.811 | n. obs.: 81 | Heteroscedasticity robust standard errors | | | |

Table C.18: Association between news and tail risk, equity markets, normal GDP regime

| Target variable: $TR_e$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 2.4965 | 1.133 | 2.204 | 0.028 | 0.276 | 4.717 |
| \| News CPI neg. \| | 1.3217 | 2.863 | 0.462 | 0.644 | -4.290 | 6.934 |
| \| News GDP pos. \| | 6.0009 | 5.160 | 1.163 | 0.245 | -4.112 | 16.114 |
| \| News GDP neg. \| | 2.0084 | 5.287 | 0.380 | 0.704 | -8.354 | 12.370 |
| \| News U pos. \| | 2.6617 | 3.787 | 0.703 | 0.482 | -4.760 | 10.084 |
| \| News U neg. \| | 1.9410 | 2.169 | 0.895 | 0.371 | -2.311 | 6.193 |
| $R^2$: 0.546 | Adj. $R^2$: 0.496 | n. obs.: 81 | Heteroscedasticity robust standard errors | | | |

## C.6 Detailed Results - Bond Markets, CPI Regimes

### C.6.1 High CPI Regime

Table C.19: Association between news and market volatility, bond markets (2-year maturity), high CPI regime

| Target variable: $RV_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0008 | 0.008 | 0.101 | 0.920 | -0.015 | 0.017 |
| \| News CPI neg. \| | -0.0069 | 0.008 | -0.862 | 0.389 | -0.023 | 0.009 |
| \| News GDP pos. \| | 0.0730 | 0.039 | 1.849 | 0.064 | -0.004 | 0.150 |
| \| News GDP neg. \| | 0.0014 | 0.021 | 0.066 | 0.947 | -0.039 | 0.042 |
| \| News U pos. \| | 0 | 0 | nan | nan | 0 | 0 |
| \| News U neg. \| | 0.0242 | 0.012 | 2.042 | 0.041 | 0.001 | 0.047 |
| $R^2$: 0.830 | Adj. $R^2$: 0.802 | n. obs.: 33 | Heteroscedasticity robust standard errors | | | |

Table C.20: Association between news and market volatility, bond markets (5-year maturity), high CPI regime

| Target variable: $RV_{b,5y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | -0.0378 | 0.027 | -1.418 | 0.156 | -0.090 | 0.014 |
| \| News CPI neg. \| | -0.0383 | 0.023 | -1.668 | 0.095 | -0.083 | 0.007 |
| \| News GDP pos. \| | 0.1472 | 0.118 | 1.245 | 0.213 | -0.085 | 0.379 |
| \| News GDP neg. \| | -0.0155 | 0.052 | -0.299 | 0.765 | -0.117 | 0.086 |
| \| News U pos. \| | 0 | 0 | nan | nan | 0 | 0 |
| \| News U neg. \| | 0.0876 | 0.039 | 2.248 | 0.025 | 0.011 | 0.164 |
| $R^2$: 0.715 | Adj. $R^2$: 0.655 | n. obs.: 33 | Heteroscedasticity robust standard errors | | | |

Table C.21: Association between news and market volatility, bond markets (10-year maturity), high CPI regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | -0.0410 | 0.036 | -1.126 | 0.260 | -0.112 | 0.030 |
| \| News CPI neg. \| | -0.0499 | 0.037 | -1.335 | 0.182 | -0.123 | 0.023 |
| \| News GDP pos. \| | 0.2627 | 0.179 | 1.465 | 0.143 | -0.089 | 0.614 |
| \| News GDP neg. \| | -0.0118 | 0.087 | -0.136 | 0.892 | -0.182 | 0.158 |
| \| News U pos. \| | 0 | 0 | nan | nan | 0 | 0 |
| \| News U neg. \| | 0.1311 | 0.057 | 2.309 | 0.021 | 0.020 | 0.242 |
| $R^2$: 0.799 | Adj. $R^2$: 0.733 | n. obs.: 33 | Heteroscedasticity robust standard errors | | | |

### C.6.2 Low CPI Regime

Table C.22: Association between news and market volatility, bond markets (2-year maturity), low CPI regime

| Target variable: $RV_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI neg. \| | -0.0161 | 0.015 | -1.075 | 0.283 | -0.045 | 0.013 |
| \| News GDP pos. \| | 0.0334 | 0.007 | 4.882 | 0.000 | 0.020 | 0.047 |
| \| News GDP neg. \| | 0.0115 | 0.014 | 0.850 | 0.395 | -0.015 | 0.038 |
| \| News U pos. \| | 0.0468 | 0.031 | 1.508 | 0.131 | -0.014 | 0.108 |
| \| News U neg. \| | 0.0250 | 0.006 | 4.519 | 0.000 | 0.014 | 0.036 |
| $R^2$: 0.70 | Adj. $R^2$: 0.660 | n. obs.: 42 | Heteroscedasticity robust standard errors | | | |

Table C.23: Association between news and market volatility, bond markets (5-year maturity), low CPI regime

| Target variable: $RV_{b,5y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI neg. \| | -0.0360 | 0.030 | -1.186 | 0.236 | -0.095 | 0.023 |
| \| News GDP pos. \| | 0.0789 | 0.015 | 5.352 | 0.000 | 0.050 | 0.108 |
| \| News GDP neg. \| | 0.0342 | 0.031 | 1.106 | 0.269 | -0.026 | 0.095 |
| \| News U pos. \| | 0.1268 | 0.063 | 2.005 | 0.045 | 0.003 | 0.251 |
| \| News U neg. \| | 0.0521 | 0.012 | 4.296 | 0.000 | 0.028 | 0.076 |
| $R^2$: 0.691 | Adj. $R^2$: 0.649 | n. obs.: 42 | Heteroscedasticity robust standard errors | | | |

Table C.24: Association between news and market volatility, bond markets (10-year maturity), low CPI regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI neg. \| | -0.0657 | 0.043 | -1.511 | 0.131 | -0.151 | 0.020 |
| \| News GDP pos. \| | 0.1629 | 0.026 | 6.207 | 0.000 | 0.111 | 0.214 |
| \| News GDP neg. \| | 0.0695 | 0.048 | 1.448 | 0.148 | -0.025 | 0.164 |
| \| News U pos. \| | 0.2607 | 0.102 | 2.550 | 0.011 | 0.060 | 0.461 |
| \| News U neg. \| | 0.0835 | 0.018 | 4.598 | 0.000 | 0.048 | 0.119 |
| $R^2$: 0.767 | Adj. $R^2$: 0.735 | n. obs.: 42 | Heteroscedasticity robust standard errors | | | |

### C.6.3 Normal CPI Regime

Table C.25: Association between news and market volatility, bond markets (2-year maturity), normal CPI regime

| Target variable: $RV_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0069 | 0.006 | 1.165 | 0.244 | -0.005 | 0.019 |
| \| News CPI neg. \| | 0.0112 | 0.018 | 0.624 | 0.533 | -0.024 | 0.046 |
| \| News GDP pos. \| | 0.0102 | 0.013 | 0.785 | 0.433 | -0.015 | 0.036 |
| \| News GDP neg. \| | 0.0088 | 0.028 | 0.319 | 0.750 | -0.045 | 0.063 |
| \| News U pos. \| | 0.0719 | 0.063 | 1.140 | 0.254 | -0.052 | 0.196 |
| \| News U neg. \| | 0.0221 | 0.006 | 3.702 | 0.000 | 0.010 | 0.034 |
| $R^2$: 0.811 | Adj. $R^2$: 0.716 | n. obs.: 52 | Heteroscedasticity robust standard errors | | | |

*20 basis point buffer to each extreme regime*

Table C.26: Association between news and market volatility, bond markets (5-year maturity), normal CPI regime

| Target variable: $RV_{b,5y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0155 | 0.017 | 0.910 | 0.363 | -0.018 | 0.049 |
| \| News CPI neg. \| | 0.0356 | 0.041 | 0.878 | 0.380 | -0.044 | 0.115 |
| \| News GDP pos. \| | 0.0160 | 0.035 | 0.457 | 0.648 | -0.053 | 0.085 |
| \| News GDP neg. \| | 0.0248 | 0.081 | 0.305 | 0.760 | -0.134 | 0.184 |
| \| News U pos. \| | 0.1808 | 0.204 | 0.887 | 0.375 | -0.219 | 0.580 |
| \| News U neg. \| | 0.0409 | 0.011 | 3.712 | 0.000 | 0.019 | 0.062 |
| $R^2$: 0.737 | Adj. $R^2$: 0.703 | n. obs.: 52 | Heteroscedasticity robust standard errors | | | |

*20 basis point buffer to each extreme regime*

Table C.27: Association between news and market volatility, bond markets (10-year maturity), normal CPI regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0324 | 0.025 | 1.276 | 0.202 | -0.017 | 0.082 |
| \| News CPI neg. \| | 0.0752 | 0.059 | 1.276 | 0.202 | -0.040 | 0.191 |
| \| News GDP pos. \| | 0.0473 | 0.052 | 0.908 | 0.364 | -0.055 | 0.150 |
| \| News GDP neg. \| | 0.0316 | 0.124 | 0.255 | 0.799 | -0.211 | 0.274 |
| \| News U pos. \| | 0.2868 | 0.349 | 0.822 | 0.411 | -0.397 | 0.970 |
| \| News U neg. \| | 0.0705 | 0.019 | 3.744 | 0.000 | 0.034 | 0.107 |
| $R^2$: 0.767 | Adj. $R^2$: 0.735 | n. obs.: 52 | Heteroscedasticity robust standard errors | | | |

*20 basis point buffer to each extreme regime*

## C.7 Detailed Results - Bond Markets, GDP Regimes

### C.7.1 High GDP Regime

Table C.28: Association between news and market volatility, bond markets (2-year maturity), high GDP regime

| Target variable: $RV_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0130 | 0.005 | 2.653 | 0.008 | 0.003 | 0.023 |
| \| News CPI neg. \| | 0.0142 | 0.008 | 1.865 | 0.062 | -0.001 | 0.029 |
| \| News GDP pos. \| | 0.0250 | 0.028 | 0.890 | 0.373 | -0.030 | 0.080 |
| \| News GDP neg. \| | 0.0034 | 0.024 | 0.145 | 0.885 | -0.043 | 0.050 |
| \| News U neg. \| | 0.0156 | 0.009 | 1.734 | 0.083 | -0.002 | 0.033 |
| $R^2$: 0.783 | Adj. $R^2$: 0.747 | n. obs.: 35 | Heteroscedasticity robust standard errors | | | |

Table C.29: Association between news and market volatility, bond markets (5-year maturity), high GDP regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0162 | 0.014 | 1.135 | 0.257 | -0.012 | 0.044 |
| \| News CPI neg. \| | 0.0358 | 0.020 | 1.825 | 0.068 | -0.003 | 0.074 |
| \| News GDP pos. \| | 0.0327 | 0.060 | 0.546 | 0.585 | -0.085 | 0.150 |
| \| News GDP neg. \| | 0.0015 | 0.052 | 0.028 | 0.977 | -0.099 | 0.102 |
| \| News U neg. \| | 0.0335 | 0.024 | 1.394 | 0.163 | -0.014 | 0.081 |
| $R^2$: 0.641 | Adj. $R^2$: 0.581 | n. obs.: 35 | Heteroscedasticity robust standard errors | | | |

Table C.30: Association between news and market volatility, bond markets (10-year maturity), high GDP regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0242 | 0.021 | 1.161 | 0.246 | -0.017 | 0.065 |
| \| News CPI neg. \| | 0.0486 | 0.032 | 1.505 | 0.132 | -0.015 | 0.112 |
| \| News GDP pos. \| | 0.0834 | 0.111 | 0.753 | 0.452 | -0.134 | 0.301 |
| \| News GDP neg. \| | 0.0079 | 0.096 | 0.082 | 0.934 | -0.181 | 0.197 |
| \| News U neg. \| | 0.0710 | 0.039 | 1.797 | 0.072 | -0.006 | 0.148 |
| $R^2$: 0.731 | Adj. $R^2$: 0.687 | n. obs.: 35 | Heteroscedasticity robust standard errors | | | |

### C.7.2 Low GDP Regime

Table C.31: Association between news and market volatility, bond markets (2-year maturity), low GDP regime

| Target variable: $RV_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI neg. \| | -0.0161 | 0.015 | -1.075 | 0.283 | -0.045 | 0.013 |
| \| News GDP pos. \| | 0.0334 | 0.007 | 4.882 | 0.000 | 0.020 | 0.047 |
| \| News GDP neg. \| | 0.0115 | 0.014 | 0.850 | 0.395 | -0.015 | 0.038 |
| \| News U pos. \| | 0.0468 | 0.031 | 1.508 | 0.131 | -0.014 | 0.108 |
| \| News U neg. \| | 0.0250 | 0.006 | 4.519 | 0.000 | 0.014 | 0.036 |
| $R^2$: 0.700 | Adj. $R^2$: 0.660 | n. obs.: 42 | Heteroscedasticity robust standard errors | | | |

Table C.32: Association between news and market volatility, bond markets (5-year maturity), low GDP regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI neg. \| | -0.0360 | 0.030 | -1.186 | 0.236 | -0.095 | 0.023 |
| \| News GDP pos. \| | 0.0789 | 0.015 | 5.352 | 0.000 | 0.050 | 0.108 |
| \| News GDP neg. \| | 0.0342 | 0.031 | 1.106 | 0.269 | -0.026 | 0.095 |
| \| News U pos. \| | 0.1268 | 0.063 | 2.005 | 0.045 | 0.003 | 0.251 |
| \| News U neg. \| | 0.0521 | 0.012 | 4.296 | 0.000 | 0.028 | 0.076 |
| $R^2$: 0.691 | Adj. $R^2$: 0.649 | n. obs.: 42 | Heteroscedasticity robust standard errors | | | |

Table C.33: Association between news and market volatility, bond markets (10-year maturity), low GDP regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI neg. \| | -0.0657 | 0.043 | -1.511 | 0.131 | -0.151 | 0.020 |
| \| News GDP pos. \| | 0.1629 | 0.026 | 6.207 | 0.000 | 0.111 | 0.214 |
| \| News GDP neg. \| | 0.0695 | 0.048 | 1.448 | 0.148 | -0.025 | 0.164 |
| \| News U pos. \| | 0.2607 | 0.102 | 2.550 | 0.011 | 0.060 | 0.461 |
| \| News U neg. \| | 0.0835 | 0.018 | 4.598 | 0.000 | 0.048 | 0.119 |
| $R^2$: 0.767 | Adj. $R^2$: 0.735 | n. obs.: 42 | Heteroscedasticity robust standard errors | | | |

### C.7.3 Normal GDP Regime

Table C.34: Association between news and market volatility, bond markets (2-year maturity), low GDP regime

| Target variable: $RV_{b,2y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0212 | 0.006 | 3.432 | 0.001 | 0.009 | 0.033 |
| \| News CPI neg. \| | 0.0032 | 0.014 | 0.220 | 0.826 | -0.025 | 0.031 |
| \| News GDP pos. \| | 0.0406 | 0.035 | 1.154 | 0.248 | -0.028 | 0.110 |
| \| News GDP neg. \| | 0.0215 | 0.037 | 0.579 | 0.563 | -0.051 | 0.094 |
| \| News U pos. \| | 0.0051 | 0.017 | 0.301 | 0.763 | -0.028 | 0.038 |
| \| News U neg. \| | 0.0162 | 0.015 | 1.058 | 0.290 | -0.014 | 0.046 |
| $R^2$: 0.658 | Adj. $R^2$: 0.613 | n. obs.: 52 | Heteroscedasticity robust standard errors | | | |

*20 basis point buffer to each extreme regime*

Table C.35: Association between news and market volatility, bond markets (5-year maturity), low GDP regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0163 | 0.013 | 1.245 | 0.213 | -0.009 | 0.042 |
| \| News CPI neg. \| | 0.0010 | 0.034 | 0.030 | 0.976 | -0.065 | 0.067 |
| \| News GDP pos. \| | 0.0247 | 0.094 | 0.264 | 0.792 | -0.159 | 0.208 |
| \| News GDP neg. \| | 0.0459 | 0.081 | 0.567 | 0.571 | -0.113 | 0.205 |
| \| News U pos. \| | 0.0345 | 0.039 | 0.874 | 0.382 | -0.043 | 0.112 |
| \| News U neg. \| | 0.0486 | 0.041 | 1.180 | 0.238 | -0.032 | 0.129 |
| $R^2$: 0.516 | Adj. $R^2$: 0.453 | n. obs.: 52 | Heteroscedasticity robust standard errors | | | |

*20 basis point buffer to each extreme regime*

Table C.36: Association between news and market volatility, bond markets (10-year maturity), low GDP regime

| Target variable: $RV_{b,10y}$ | coef | std err | z | P> \|z\| | [0.025 | 0.975] |
|---|---|---|---|---|---|---|
| \| News CPI pos. \| | 0.0415 | 0.026 | 1.607 | 0.108 | -0.009 | 0.092 |
| \| News CPI neg. \| | 0.0005 | 0.056 | 0.009 | 0.993 | -0.110 | 0.111 |
| \| News GDP pos. \| | 0.0595 | 0.153 | 0.390 | 0.696 | -0.239 | 0.358 |
| \| News GDP neg. \| | 0.0811 | 0.143 | 0.566 | 0.572 | -0.200 | 0.362 |
| \| News U pos. \| | 0.0600 | 0.073 | 0.821 | 0.412 | -0.083 | 0.203 |
| \| News U neg. \| | 0.0808 | 0.067 | 1.205 | 0.228 | -0.051 | 0.212 |
| $R^2$: 0.543 | Adj. $R^2$: 0.483 | n. obs.: 52 | Heteroscedasticity robust standard errors | | | |

*20 basis point buffer to each extreme regime*

# D

# Appendix: Monetary Policy Shocks

## D.1 Topic Compositions

Figure D.1: Top words for CPI model with interaction terms between topics and numerical covariates (K=20)

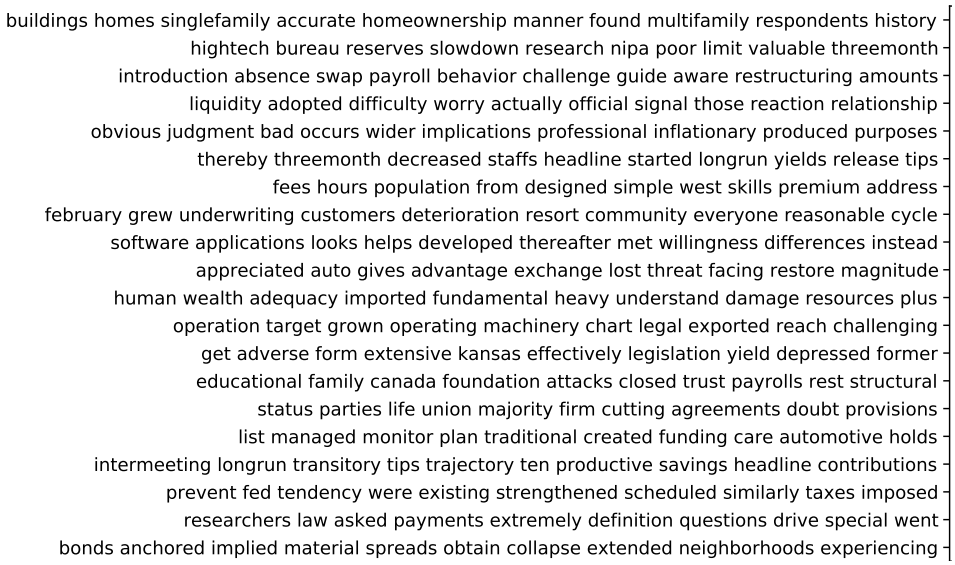


Figure D.2: Top words for GDP model with interaction terms between topics and numerical covariates (K=20)

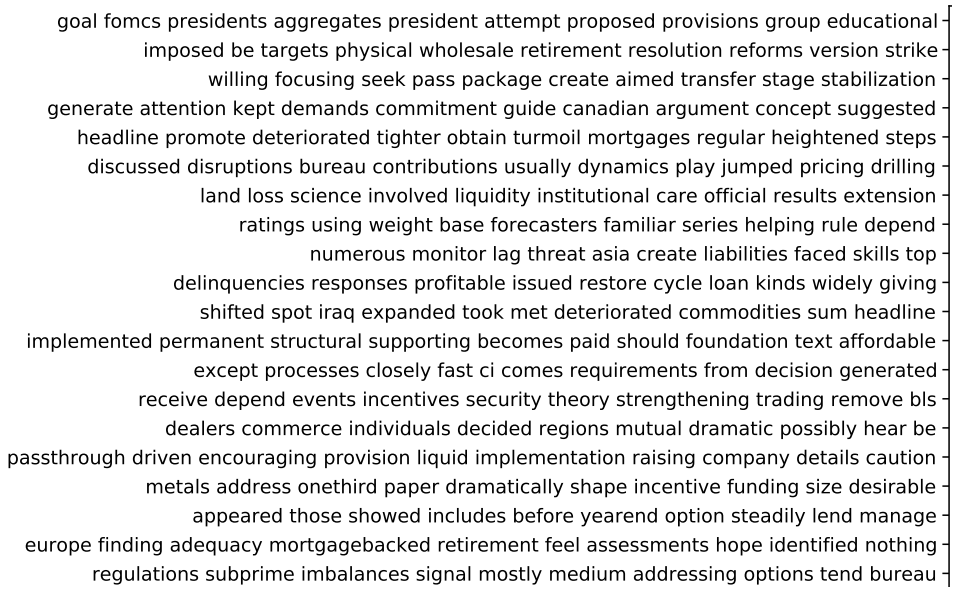
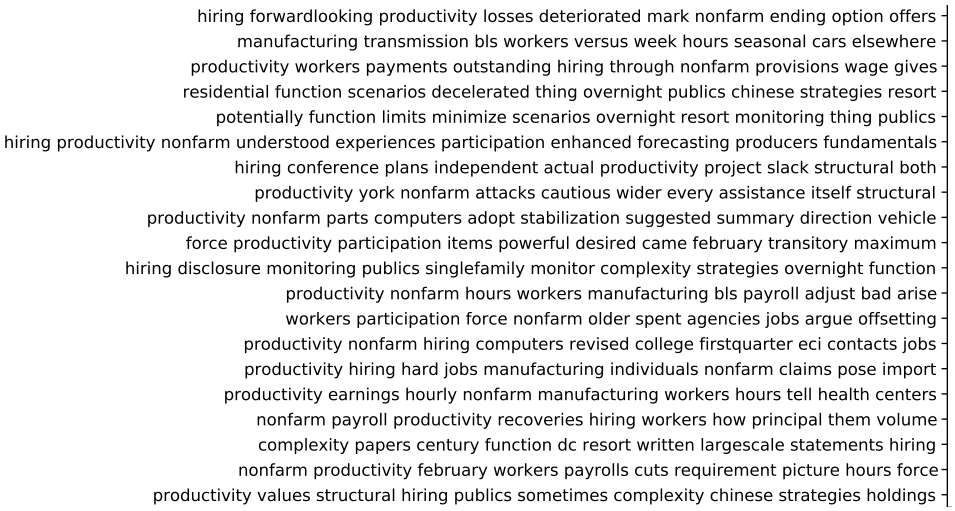



Figure D.3: Top words for Unemployment model with interaction terms between topics and numerical covariates (K=20)



## D.2 Validation Set Performance

Exemplary validation set performances for model estimation runs on targets: (1) GDP, (2) CPI, and (3) unemployment. MSE and perplexity reported. Optimal parameters correspond to epoch that yielded lowest validation set MSE. Max. epochs = 2000.

Figure D.4: CPI - validation loss

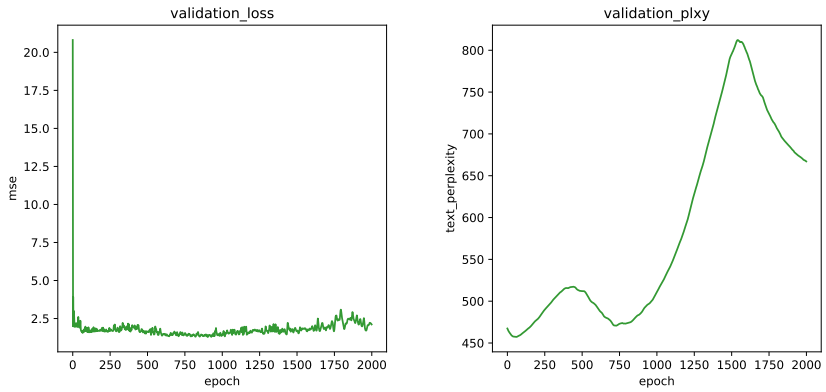



Figure D.5: Unemployment - validation loss

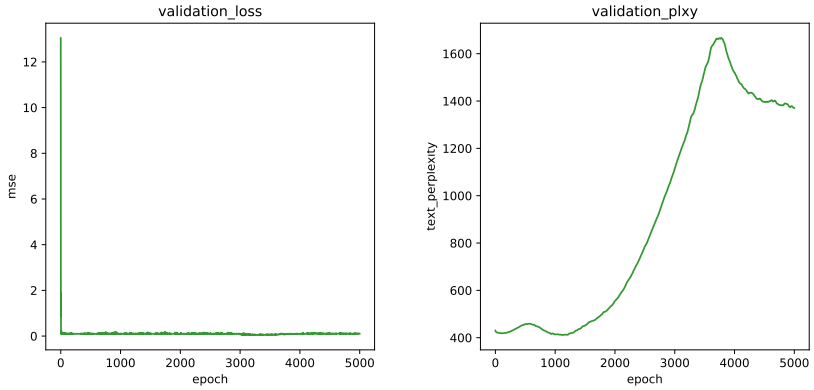
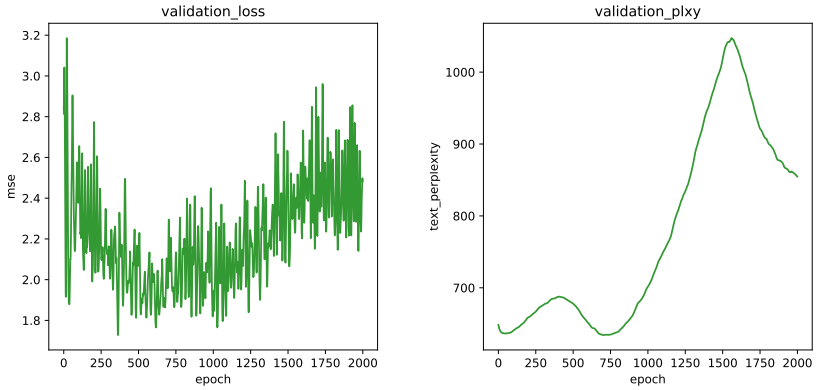


Figure D.6: GDP - validation loss



## D.3 Implied Signal Dispersions

Figure D.7: GDP speech signal by central banker

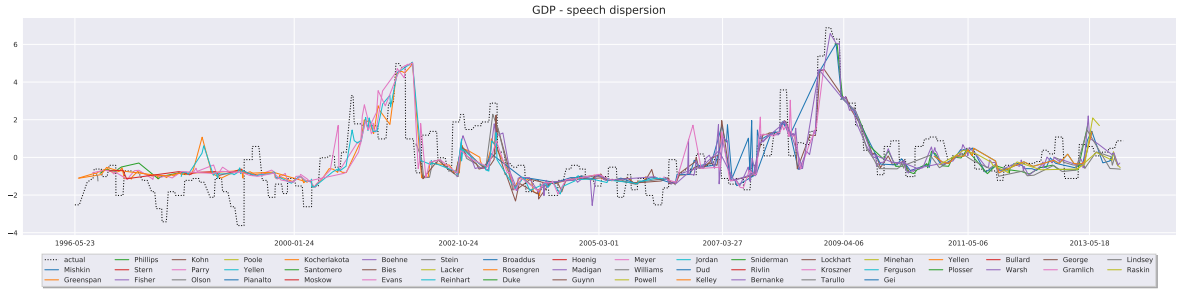



Figure D.8: CPI speech signal by central banker

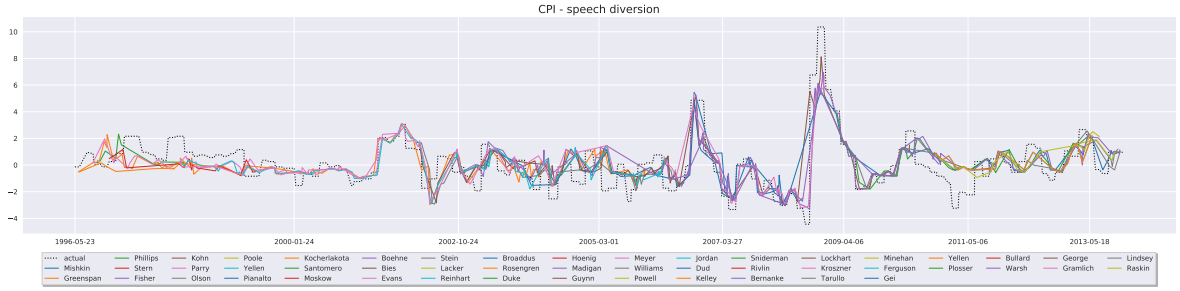



Figure D.9: Unemployment speech signal by central banker

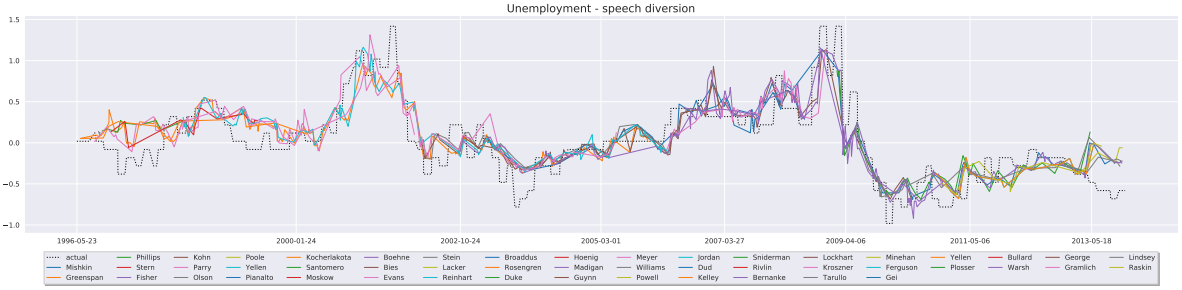
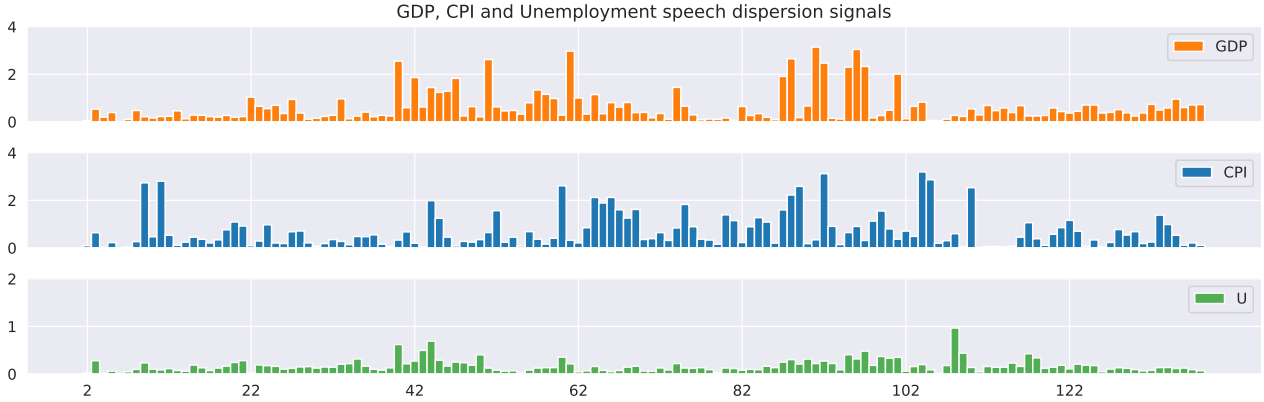



Figure D.10: Signal dispersion across speakers (grouping window: inter-FOMC-meeting periods)



## D.4 Central Bank Speech Examples

**Speech by Donald Kohn, 19th of November 2008. Estimated as indicating a very dovish stance on CPI,given all other economic indicators at that time:**

Vice Chairman Donald L. Kohn At the Cato Institute's 26th Annual Monetary Policy Conference, Washington, D.C. November 19, 2008 Monetary Policy and Asset Prices Revisited As you know, we are in the midst of a global financial crisis that is now weighing heavily on economies around the world. Although the outlook remains extremely uncertain, both the fragility of the financial system and the weakness in real activity seem likely to persist for a while. To promote maximum sustainable economic growth and price stability, the Federal Reserve has responded to this crisis by easing monetary policy markedly, and we have greatly expanded our liquidity facilities to keep credit flowing when private lenders have become reluctant or unable to do so. Other central banks have also cut policy rates significantly and expanded their lending. In addition, the federal government and governments around the world have taken extraordinary actions to strengthen financial systems to preserve the ability of households and businesses to borrow and spend.1 The current situation is so severe that it calls for careful review of how such a crisis evolved and how we can prevent a similar situation from happening again. This conference is a welcome step in that review, as it asks about the lessons we have learned, particularly for monetary policy, from the collapse of subprime lending and the preceding house-price bubble–developments that contributed importantly to the present financial crisis. This morning I would like to reflect on some of what I, in my role as a monetary policy maker, have learned from recent developments in the housing sector and, more broadly, in financial markets as a whole. In doing so, I will revisit the remarks I made in 2006 in Frankfurt at a festschrift for Otmar Issing.2 There I argued that a central bank facing a possible asset bubble would have to surmount some high hurdles before it would be justified in tightening policy beyond what the outlook for output and inflation would require, after taking into account past and projected asset price developments. In the aftermath of the collapse of the housing market and in the midst of the ensuing financial and economic turmoil, does that conclusion still hold? More time and study will be needed before we can be confident about the lessons of the current crisis. But to foreshadow the remainder of these remarks, based on what we know today, I still

have serious questions about whether trying to use monetary policy to check speculative activity on a regular, systematic basis would yield benefits that outweigh its costs. I hasten to add that it is evident from the current crisis that much has to change on the regulatory front. Governments around the world face the challenge of revamping the regulatory structure governing financial markets. And changes in this area, I believe, will prove to be the most necessary and effective at reducing the odds on another severe financial crisis. Today, however, I will focus on some of the lessons of the current crisis for monetary policy. Alternative Strategies for Addressing Asset Price Bubbles In my 2006 speech, I discussed two different strategies for monetary policy to deal with a possible asset price bubble–the "conventional strategy" and "extra action." A central bank following the conventional strategy does not attempt to use monetary policy to influence the speculative component of asset prices, on the assumption that it has little ability to do so and that any attempt will only result in sub-optimal economic performance in the medium run. Instead, the central bank responds to asset price movements, whether driven by fundamentals or not, only to the degree that those movements have implications for future output and inflation. This conventional strategy conforms to the Federal Reserve's dual mandate under the law and it has been our policy strategy; it also has been consistent with the practices of most inflation-targeting central banks. However, some observers have argued for a more activist policy than this one. Specifically, they have urged central banks, upon perceiving the development of an asset bubble, to take extra action by tightening policy beyond what the conventional strategy would suggest, with the hope of limiting the size of the bubble and thus the fallout from its deflation. Such a strategy, if successful, could deliver substantial benefits, and a number of central bankers have talked about the need to consider a policy of extra action on occasion, and perhaps have even implemented such a strategy. However, taking extra action also would entail some costs, such as creating, for a time, higher unemployment and lower inflation than would otherwise be desired. In assessing these two alternatives for monetary policy, in the 2006 speech I concluded that a strategy of extra action might be justified if three tough conditions were met. First, policymakers must be able to identify bubbles in a timely fashion with reasonable confidence. Second, a somewhat tighter monetary policy must have a high probability that it will help to check at least some of the speculative activity. And third, the expected improvement in future economic performance that would result from the curtailment of the bubble must be

sufficiently great. Of course, we live in an uncertain world, and accordingly policymakers should always be open to the possibility that these conditions might be satisfied and that extra action would be appropriate. But my thought at the time was that, in practice, the likelihood of ever meeting the three conditions seemed remote. In the aftermath of the bursting of the housing bubble, however, the severity of the fallout might seem to call this judgment into question. So let's re-examine each of the three conditions and see what the current crisis has taught us. Potential Gain from Limiting Bubbles Let me start with my third condition, the potential gain from limiting bubbles, because this is where my views have changed the most. Although I was concerned about the potential fallout from a collapse of the housing market, I think it is fair to say that these costs have turned out to be much greater than I and many other observers imagined. In particular, I and other observers underestimated the potential for house prices to decline substantially, the degree to which such a decline would create difficulties for homeowners, and, most important, the vulnerability of the broader financial system to these events. In retrospect, I may have been unduly comforted by the resilience of the U.S. economy to the collapse of the high-tech bubble, to the earlier Russian debt default and failure of Long-Term Capital Management, and even to the commercial and residential real estate debacles of the late 1980s and early 1990s (as difficult as that recovery was). But mopping up after this asset price bubble has turned out to be much harder because of its greater magnitude, the centrality of residential housing and finance to our economy and financial system, and the surprising ways obscure and complex financial transactions have exposed banks and other financial institutions to heavy losses. In addition, financial and economic linkages across countries have made this crisis truly global in scope, affecting both developed and developing economies. As a result of all these factors, the economic disruption here and abroad is likely to be considerably more severe than in past episodes. The severe fallout may indicate a larger potential gain than I had anticipated to leaning against excess exuberance in asset markets. However, realizing that potential rests on meeting my two other conditions as well–the timely identification of the bubble, and the ability of a central bank to materially influence the trajectory of the speculative component of asset prices. Identifying Bubbles in a Timely Manner As for the first of the three conditions, events of the past few years, coupled with advances in our understanding of how bubbles form and persist, have made me a little less dubious that policymakers can reliably

identify a serious bubble before it bursts. However, I am still skeptical about our ability to detect bubbles early enough to make a general policy of leaning against them successful on average. The identification of bubbles in real time is tricky because not all the fundamental factors driving asset prices are directly observable; thus, any judgment by a central bank that an asset is overpriced is by nature uncertain. My views on this aspect of the identification problem have been reinforced by my experience during the inflation of the housing bubble. Over the first half of the decade, we saw a sustained, rapid rise in both home values and mortgage debt. As this process continued, concern about its sustainability grew and many observers started speculating that a bubble was in place. During this period, staff throughout the Federal Reserve System examined whether house prices were overvalued and arrived at a wide range of answers. For example, one set of models that linked rental rates and house prices indicated as early as the start of 2004 that the market was significantly overvalued, while another set of models suggested, even as late as December 2005, that house prices could be justified by fundamentals.3 Thus, controversy over the existence of a bubble persisted almost right up to the actual peak in the housing market. Because the economic consequences of mistakenly responding to a misidentified bubble are substantial, central bankers may be reluctant to take extra action in the face of such uncertainty, especially if they are risk-averse. Policymakers may also be reluctant to act because a bubble "call" might seem to require them to be more knowledgeable than market participants. After all, if at least some market participants perceive the emergence of a bubble, wouldn't they arbitrage that mispricing away? Recent research, however, suggests reasons for why market participants who think they know that a bubble exists still may not trade to eliminate it. For example, if some market participants recognize the presence of a bubble but do not know how common their knowledge is, they might reasonably expect to make the most profits by riding the bubble for as long as possible, with the goal of trying to sell the asset just before it collapses.4 Other research emphasizes that certain institutional structures–such as secured lending and delegated portfolio management–can create substantial costs in trading against an asset price bubble, so that even market participants who are conscious of the bubble will not find it profitable to trade against it.5 Together, these studies suggest that policymakers may be able to detect bubbles that will not be quickly arbitraged away, thus strengthening the argument for considering extra action.6 Nonetheless, even if policymakers are confident that a

bubble has emerged, the question of the timeliness of the call remains. The essential problem is the timing of the detection of the bubble relative to the timing of its collapse. The risk is that the detection and subsequent policy response occur not long before the bubble collapses on its own. Given the lags associated with monetary policy, the resulting contractionary effects on the economy of the monetary tightening would occur just when the adverse effects of the bubble's collapse are being realized, worsening rather than mitigating the effects of the bubble's collapse. And the inevitable lags in detecting bubbles increase the likelihood that, by the time action is taken, speculative activity will have progressed to the point that its collapse is not far off. Thus, even if we could have known for sure that a housing bubble existed, and that tighter monetary policy would have significantly checked the unwarranted rise in home prices, policymakers would have had to make this call early on–at least a year and probably more before the peak in the real estate market in 2006–for such an action to have been beneficial. Ability of Monetary Policy to Influence Bubbles This brings me to the remaining condition–the requirement that monetary policy be able to materially check expansions in asset bubbles. Clearly, interest rates play an important role in determining the fundamental value of corporate equity, houses, and other assets. However, I noted in my earlier speech that the influence of interest rates on the speculative component of asset prices is unclear from both a theoretical and empirical standpoint. My views on this issue have not changed much, largely because of the still-murky role that monetary policy played in promoting the surge in house prices and the accompanying run-up in both conventional and subprime mortgage debt. Although tighter monetary policy might have succeeded in shifting down the path of house prices, it is still not clear to what extent small or even moderate policy actions would have discouraged the broader speculative developments that have characterized the current episode: overly optimistic expectations of price appreciation, excessive leveraging, and a marked increase in risk-taking by homeowners and investors. Of course, a substantial tightening of policy, leading to a significant slowing in the economy and rise in unemployment, might have had a marked effect on housing price gains. But undertaking such a policy course on a regular basis whenever asset price misalignments are detected would likely prove to be a relatively poor strategy on average, especially given the possibility of false positives in identifying these misalignments, and the existence of other potential remedies. In general, taking more-targeted steps–for example, regulatory changes intended to strengthen the

financial system–would seem a better course of action under such circumstances. To be sure, some observers contend that the low level of the federal funds rate in 2003 and 2004 was clearly a primary cause of the housing bubble, and that a significantly tighter stance of monetary policy would have been warranted. As you know, the Federal Open Market Committee (FOMC), after having sharply lowered its policy rate during the 2001 recession, further lowered the federal funds rate in late 2002 and 2003 in response to an outlook for continued tepid real growth and a possible unwelcome disinflation. This accommodative stance helped set the stage for a more robust recovery, and as the expansion took hold in 2004, the FOMC began to tighten in a gradual manner that was publicly signaled in advance. How might these monetary policy actions have fueled speculation? Perhaps a low policy rate early in the decade, by stimulating housing demand and pushing up the level of home prices, incorrectly led households and lenders to extrapolate these price increases into the indefinite future. Overly optimistic expectations may have had an unusually stimulative effect on the housing market after 2003 because borrowing constraints were being eased by new financial developments, such as the growth of subprime lending and other nontraditional mortgages, fueled in part by investor demands for the higher yields on complex structured products.7 In addition, the increased use of adjustable-rate mortgages–which are more closely tied to short-term policy rates–may have initially boosted the stimulus from a lower federal funds rate. These stories have a certain plausibility, but a closer examination raises questions about monetary policy and the housing and credit bubbles. Although low short-term interest rates probably supported housing demand and home prices for a time–an effect that helped offset the negative effects on economic growth and employment of the steep decline in business investment–the role of monetary policy in fueling the speculation in real estate is still not clear. Studies that have tried to address how much monetary policy contributed to the increase in house prices during this period are inconclusive.8 And in general, the channel from interest rates to house prices has not been strongly established empirically, suggesting it might take a very large hike in the federal funds rate to have a substantial effect on real estate values.9 Moreover, if accommodative monetary policy engendered extrapolative expectations and speculation starting in 2003, why did it not restrain these factors after mid-2004 as the federal funds rate was increased? Tightening should have limited the extent to which households (especially those using variable-rate mortgages) were able to borrow, thereby slowing the pace

of house price appreciation. Furthermore, many of the worst subprime loans were made after the federal funds rate had normalized, and reflected a wide array of deficiencies in the financial markets. The contrasting movement of short-run and long-term interest rates over this period further complicates any assessment of the link between monetary policy and the housing market. Housing demand and home prices are, presumably, most closely linked to the 30-year fixed mortgage rate and the expected average borrowing rate to be paid over the life of adjustable-rate mortgages. That these actual and expected loan rates moved sideways even as the federal funds rate rose suggests that other factors besides monetary policy were at work, especially since the FOMC clearly signaled that it would be returning the funds rate to a normal level over time (albeit at a "measured pace"). A good portion of the appreciation in house prices probably is due to the structural changes that were taking place in mortgage financing–specifically, the opening up of subprime lending and the expansion in associated securitization markets with its strong demand for mortgages from investors. Gauging the effects of expanded subprime lending on house prices is complicated by two-way causality–more lending can drive up house prices, but expected house price increases can also induce more lending. Undoubtedly, causality did indeed run both directions. But studies do indicate that an expansion in credit leads to increased house prices, and suggest that structural changes in mortgage finance likely boosted the rate of house price appreciation.10 Another key observation that must be reconciled with any explanation of recent events is that the run-up and subsequent decline in house prices was not limited to the United States; indeed, some countries have experienced even larger swings in house prices.11 In most countries during this period, long-term interest rates were low despite the fact that their central banks did not ease monetary policy as markedly as the Federal Reserve. A common factor behind these low rates, and perhaps in part behind the shared increase in house prices as well, is the "global saving glut" identified by Chairman Bernanke–the large amounts of savings, both official and private, from Asian and oil-exporting nations that tended to lower neutral interest rates globally.12 In a broader sense, perhaps the underlying cause of the current crisis was complacency. With the onset of the "Great Moderation" back in the mid-1980s, households and firms in the United States and elsewhere have enjoyed a long period of reduced output volatility and low and stable inflation. These calm conditions may have led many private agents to become less prudent and to underestimate the risks associated with their actions. While we

cannot be sure about the ultimate sources of the moderation, many observers believe better monetary policy here and abroad was one factor; if so, central banks may have accidentally contributed to the current crisis. But would a somewhat tighter stance of policy in recent years have reversed this complacency? It seems doubtful. Central banks would likely have needed to produce recessions of some consequence in order to force agents to reevaluate the costs of taking on risk–an outcome unlikely to improve societal welfare. Rather than using the blunt tool of monetary policy to induce prudence, we should examine more closely the possibility of using regulation and prudential supervision to address concerns about overleveraging and other risk-taking behavior. In short, we still do not fully know what caused the run-up in house prices and over-building. Short-term rates were low in 2002-04 as the Federal Reserve countered the risks it saw to good economic performance, and these low rates probably had some effect on housing markets at the time. But the problems largely built up after policy rates were well on their way to neutral, and other factors appear to have played major roles. We have learned little about the likely effect that a somewhat higher funds rate would have had on the speculative element of prices. Of course, it is important to keep an open mind about the relationship of short-term interest rates and speculative activity. If it becomes clear that monetary policy can predictably influence the evolution of bubbles, central banks should take that ability into account when crafting policies intended to keep output rising in line with its potential and inflation low and stable. Conclusion In sum, I am not convinced that the events of the past few years and the current crisis demonstrate that central banks should switch to trying to check speculative activity through tighter monetary policy whenever they perceive a bubble forming. The recent experience may have made us a bit more confident about detecting bubbles, but it has not resolved the problem of doing so in a timely manner. Nor has it shown that small-to-modest policy actions will reliably and materially damp speculation. For these reasons, the case for extra action still remains questionable, despite our having learned that the aftermath of a bubble can be far more painful than we imagined. Some may object to this assessment, arguing that the current crisis is so bad that, in retrospect, monetary policy should have been appreciably tighter to deflate or forestall the housing boom earlier in the decade, even if that meant a substantially weaker economy. This argument has two defects. First, monetary policy is made in real time, not with the benefit of hindsight, and any evaluation of competing

strategies for the systematic conduct of policy must be grounded in that fact. Although we must learn from history, we cannot implement policy strategies that assume more information about the future than we can ever have. Second, even if we ignore the fact that policymakers at the time could not have known what the future held in store if the funds rate followed the path it actually did, we also need to recognize that we cannot be sure what would have happened if policy had taken a different course. If policy had tightened appreciably at an early stage of the housing boom, say in mid-2003, it would have done so when the unemployment rate was still rising and inflation seemed poised to move to an undesirably low level. Such a course of action might well have created its own unforeseen consequences that we might now be ruing. This assessment aside, recent events would seem to have some implications for the conduct of monetary policy. For example, in light of the demonstrated importance to the real economy of speculative booms and busts (which can take years to play out), central banks probably should always try to look out over a long horizon when evaluating the economic outlook and deliberating about the appropriate accompanying path of the policy rate. The Federal Reserve staff has for sometime regularly provided the FOMC with this sort of extended-horizon analysis. In particular, the staff regularly generates likely paths for the economy over the next five years or so under different economic and policy assumptions; these scenarios often highlight different possibilities for the evolution of prices for homes and other assets. Note that the focus here is not a single baseline outlook; rather, the emphasis is on exploring the various ways events could play out and the implications for monetary policy. Another lesson of the current crisis is that central banks need to improve their understanding of the workings of the financial system, its vulnerabilities, and its links to the real economy. We must try to find ways to discern more quickly if financial innovation and other factors are leading to a buildup of destabilizing forces, such as rapidly rising asset prices or excessive leverage. Moreover, the unexpectedly rapid resonance of financial turmoil through global markets signals a need for further study of the complex cross-country linkages among lenders and borrowers, and the ways in which those linkages are influenced by such factors as leverage, interdependent counterparty relationships, and backup liquidity agreements. Finally, more effort needs to be spent on further investigation of the financial accelerator and other credit-channel effects, given the accumulating evidence that such effects can give rise to an adverse feedback loop between financial markets and the

real economy. Overcoming these deficiencies in our knowledge will not be easy, but the potential benefits could be great. Finally, as I emphasized at the outset, we must thoroughly review the regulatory structure of the U.S. and global financial systems, with the objective of both identifying and implementing the comprehensive changes needed to reduce the odds of future bubbles arising, and improving the ability of banks and other financial institutions to weather the fallout from unexpected adverse changes in asset prices. Ultimately, this process should prove our best line of defense against the problems of the sort we now face.

**Speech by Jeffrey M. Lacker, 3rd of December 2008. Estimated as indicating a very hawkish stance on CPI, given all other economic indicators at that time**:

These are economically trying times. In my remarks, I would like to discuss the factors I see affecting the outlook for the U.S. economy and monetary policy. As always, I speak only for myself, and not for my Federal Reserve System colleagues.1 Financial market conditions loom large in any discussion of the economy these days. The heart of the problem, of course, is the home mortgages made from late 2005 through early 2007, near the end of the long U.S. housing boom that began in 1995. Since the peak in activity in 2005, housing investment has fallen by more than 40 percent. Average housing prices, as measured by the FHFA repeat sales index, have fallen 6 percent since their peak in April 2007. Some markets have experienced more dramatic declines; the home price index for California fell 18 percent, for example. The resulting erosion in home equity for many borrowers has meant that mortgages made near the peak of the boom, especially the subprime and non-traditional categories, are experiencing much larger losses than expected. It will take years of research to untangle the quantitative contribution of various causal factors to the rise in subprime mortgage lending and the increase in subprime losses, so I wont attempt such an analysis here. Let me simply offer a list of plausible suspects. One candidate is the wave of technological innovation in retail credit delivery, which contributed to an expansion of consumer credit, including unsecured and mortgage credit. As in any industry in the midst of innovation, this expansion may have involved overshooting and retrenchment. A second suspect is the regulatory and supervisory framework surrounding U.S. housing finance, which may have been insufficiently prepared for the possibility of a swing in housing demand of the magnitude and geographic extent that we have seen. Private sector incentives to foresee and protect against such shocks were to some extent dampened by the presence of the federal financial safety net, and perhaps by official policies aimed at increasing homeownership. In addition, the unscrupulous and fraudulent practices of some mortgage brokers outside of the banking sector may have contributed to the problem. I would also cite relatively low interest rates after the recession earlier this decade, especially in 2003 and 2004. Some economists have argued, with the benefit of hindsight, that tighter monetary policy during that period would have led to better outcomes by preventing core inflation from rising, thus limiting the housing

boom and mitigating the subsequent bust.2 While I find this view plausible, again, further research will be required to substantiate this hypothesis. That's all prologue, however, to the turmoil that has plagued financial markets since the middle of last year, when the potential scale of the home mortgage problem became more widely appreciated. The turmoil intensified in mid-September this year, and volatility has been elevated since. Financial market participants have faced three major categories of uncertainty. The first concerns the aggregate amount of losses on mortgage lending. For mortgages made in 2006 and early 2007 the vintages in which losses are concentrated significant uncertainty still remains regarding total losses. Second, financial market participants face uncertainty about where the losses will turn up. Mortgage risks were split up and spread widely, both within the United States and in Europe, through securitization and use of the insurance capabilities provided by credit derivative contracts. As a result, financial market participants are understandably apprehensive about whether a particular counterpartys mortgage-related losses will erode their capital buffer enough to threaten their viability. This has led to elevated risk premia in interbank credit markets for institutions with at least some presumed mortgage-related exposure. Third, market participants have at times faced uncertainty about prospective public sector intervention.3 The disparate responses to potential failures at several high-profile organizations this year may have made it difficult for market participants to forecast whether and in what form official support would be forthcoming for a given counterparty. Shifts in expectations regarding official intervention may have added volatility to financial asset markets that already were roiled by an increasingly uncertain growth outlook. The striking feature of central bank lending during the recent turmoil is the extent to which it has extended well beyond the boundaries that previously were understood to constrain such lending, both in the range of institutions and the contractual terms on which credit has been provided. Intervention has been driven by a desire to prevent damaging disruptions to financial markets, and thus reduce the overall costs of the turmoil. While this objective is clearly understandable, central bank lending can create the expectation that similar support will be forthcoming when market disruptions occur in the future. Such expectations can themselves be very costly, because they can distort the incentives faced by, and as a result, the choices made by private-sector participants. The critical policy question of our time is where to establish the boundaries around the public-sector safety net provided to financial market participants, now

that the old boundaries are gone. In doing so, the prime directive should be that the extent of regulatory and supervisory oversight should be commensurate with the extent of access to central bank credit in order to contain moral hazard effectively. The dramatic recent expansion in Federal Reserve lending, and government support more broadly, has extended public sector support beyond existing supervisory reach, and thus could destabilize the financial system, if no corrective action is taken. Restoring consistency between the scope of government support and the scope of government supervision is essential to a healthy and sustainable financial system. One option is simply to adapt our regulatory and supervisory regime to the new wider implied reach of government lending support. This strikes me as an unattractive option, if for no other reason than the current uncertainty about the outer bounds of that support. Constraining moral hazard in such a regime would be an immense and daunting task. I take it as given, therefore, that the scope of financial safety net ultimately must be rolled back. Note that it will not be sufficient simply to roll back the current lending programs when the economy recovers. The precedents that have been set during this episode will influence how market participants expect policymakers to react during the next episode of financial market turmoil. Establishing a coherent and stable financial regulatory regime will require rolling back expectations about how the policymakers will respond to the next financial market disturbance. Rolling back those expectations will be impossible if moral hazard concerns are always set aside in the exigencies of a crisis.4 Assessing the effects of financial market turmoil on real economic spending is not as straightforward as it might seem. One popular notion is that the credit market disruptions weve seen over the last year or so impede the financial sectors ability and willingness to extend credit to households and business firms, thereby creating an additional drag on spending. But causation can flow in the opposite direction as well. When overall economic activity seems poised to contract, the outlook for household income and business revenues deteriorates as well, and such borrowers become less creditworthy, all else constant. My reading of current conditions is that bank lending is constrained more now by the supply of creditworthy borrowers than by the supply of bank capital. The decline in U.S. housing activity since early 2006 has affected not only credit markets  it has had a significant impact on broader economic activity as well. For a time, the weakness was isolated in the housing market, as the rest of the economy continued to expand at a relatively healthy rate. But late last year, consumer spending began to slow.

Household net worth has declined as home prices have fallen virtually nationwide over the last year-and-a-half, and, more recently, equity prices have slumped. Increases in energy prices up through the middle of this year took a substantial bite out of real incomes. Moreover, payroll employment peaked last December, and has since shed 1.2 million jobs. As the labor market has weakened, wage growth has tapered off. Except for the temporary bulge due to the stimulus payments earlier this year, real personal income has steadily decelerated, and is now below where it was a year ago. Given this catalog of adverse developments for U.S. households, it should be no surprise that consumer spending was sluggish in the first half of the year and has fallen significantly in recent months. When household spending slows substantially, business capital investment is usually not far behind. Business spending on equipment and software fell in the first half of 2008, and the near-term outlook is not favorable. Many firms are facing dimmer sales prospects, higher funding costs, and more restrictive borrowing terms. The other segment of business fixed investment, spending on new structures, has been booming recently. In 2007 and the first half of 2008, real nonresidential fixed investment  a segment that includes office buildings, hotels, malls and the like  grew at a 14 percent annual rate. That category seems to have topped out over the summer, and is certain to decline in coming months. Foreign trade has added significantly to GDP growth last year and the first half of this year. Unfortunately, the trade contribution to U.S. growth is likely to decline in the near term in response to diminishing world growth prospects and the recent strength in the dollar. Two days ago, the National Bureau of Economic Research officially confirmed what virtually all economists already knew  namely, that a recession began last December when payroll employment peaked. For a time, the decline was fairly mild  in fact milder than the last two recessions, both of which were themselves mild by historic standards. But conditions downshifted dramatically sometime in September, just as financial market turmoil was accelerating. Since then, according to reports, many households and firms are taking a wait and see attitude, reducing or postponing nonessential outlays in response to a general sense of uncertainty about the potential meaning of these dramatic events for their own economic circumstances. A wide array of economic indicators has deteriorated markedly since then as well. Looking ahead, uncertainty about the outlook is greater than usual, though probably not greater than is typical for this phase of a business slowdown. It strikes me as reasonable to expect the U.S. economy to regain positive momentum sometime in 2009,

for several reasons. First, monetary policy is now quite stimulative. Second, the energy and commodity price shocks that dampened economic activity earlier this year have subsided already or are in the process of doing so. And as Ive mentioned, the drag from housing seems likely to lessen in the next year, and in fact, I would be surprised if we dont see a bottom in housing construction sometime in 2009. This is the third straight year, however, that Ive been expecting a bottom in the housing market in the middle of next year, so my outlook is tempered by more than the usual amount of humility. While the downturn in real economic activity is going to pose challenges for monetary policy in the period ahead, its essential that we not let inflation drift from view. Since 2004, overall inflation has trended upward, and has been higher than I would like, over the last few years. Much of the acceleration we saw earlier this year reflected energy prices, however, and with oil prices down we have seen overall inflation subside in recent months. Many economists are forecasting relatively low inflation in the months ahead, on the grounds that widening economic slack is generally associated with declining price pressures. While this correlation is detectable in many datasets, I would be cautious about relying on it as a causal relationship.5 And while it may seem premature to be worrying about how inflation behaves after the recession is over, we need to be sure our policy remains consistent with a strategy that does not allow inflation to ratchet up over the business cycle. As I said at the outset, these are not the best of economic times. We have weathered economic downturns before, however, both nationally and globally. And there is no sign that the fundamental creative process that drives innovation and improves well-being over time has been mortally wounded. What sets this episode apart is the nature of the turmoil plaguing the financial sector, and the array of unprecedented government lending programs. While navigating the slowdown in real economic growth is a challenge, the larger and more significant challenge will be to re-establish the boundaries around central bank lending and public sector support and reconstruct the relationship between the public sector and financial markets. How well we meet this challenge will determine the extent to which innovation, despite the associated volatility, will continue to contribute to the effectiveness of our financial system and to overall economic growth.