Making text count: economic forecasting using newspaper text[*]

Eleni Kalamara[†] Arthur Turrell[‡] Chris Redl[§] George Kapetanios[¶] Sujit Kapadia[‖]

January 16, 2022

Abstract

This paper examines several ways to extract timely economic signals from newspaper text and shows that such information can materially improve forecasts of macroeconomic variables including GDP, inflation, and unemployment. Our text is drawn from three popular UK newspapers that collectively represent UK newspaper readership in terms of political perspective and editorial style. Exploiting newspaper text can improve economic forecasts both unconditionally and when conditioning on other relevant information, but the performance of the latter varies according to the method used. Incorporating text into forecasts by combining counts of terms with supervised machine learning delivers the highest forecast improvements relative to existing text-based methods. These improvements are most pronounced during periods of economic stress when, arguably, forecasts matter most.

Keywords: text, forecasting, machine learning

JEL Codes: C53, C82, C45

[*] First version: September 10, 2019. This version: January 16, 2022. The views in this work are those of the authors and do not represent the views of the Office for National Statistics, the European Central Bank, the International Monetary Fund, or the Bank of England, where CR, AT, and SK were employed when the project began. We are grateful to David Bholat, Paul Robinson, Misa Tanaka, and to conference participants at the Federal Reserve Board of Governors, the European Central Bank, the Bank of England, Danmarks Nationalbank, the European Commission, the 2019 European Economic Association meeting, the 2018 Royal Economic Society meeting, the 2019 Economic Statistics Centre of Excellence meeting, the H. O. Stekler Research Program on Forecasting Seminar at George Washington University, and the 2020 American Economics Association meeting. David Bradnum provided outstanding technical assistance throughout.

[†] King's College London. Email: eleni.kalamara@kcl.ac.uk

[‡] Office for National Statistics; prev. Bank of England. Email: arthur.turrell@ons.gov.uk

[§] International Monetary Fund; prev. Bank of England. Email: credl@imf.org

[¶] King's College London. Email:george.kapetanios@kcl.ac.uk

[‖] European Central Bank; prev. Bank of England. Email: sujit.kapadia@ecb.europa.eu

“A good newspaper, I suppose, is a nation talking to itself”

— Arthur Miller

1 Introduction

This paper shows that newspapers are informative for a nation’s economic future – and that an effective way to obtain that information is with high-dimensional text analysis methods that exploit machine learning.

We show how contemporaneous newspaper text data can enhance forecasts and inform policymaking by using a range of existing and novel methods of turning text into time series to i) extract forward looking economic indicators from text and ii) use newspaper text for economic forecasting. Our data comprise three UK daily newspapers over the period 1990 to 2019 that have high circulation and represent a broad swathe of UK newspaper readership both in terms of political perspective and editorial style.

We find that text significantly improves forecasts of macroeconomic variables, including GDP, inflation, and unemployment, relative to widely used benchmarks. This is especially true during periods of stress, suggesting that newspaper text could speak to the prediction of recessions and is a strong complement to high frequency financial market data and to more expensive, and often less timely, survey data. This relationship holds at forecasting horizons nine months into the future. Taken together, our findings suggest that newspaper content is informative for reasons beyond just its greater timeliness – either because it provides independent insights on economic developments (not captured by other sources of information) or perhaps even because it may influence ‘animal spirits’, shaping the economic behaviour of households and businesses and thus the future path of the macroeconomy (Keynes, 1936; Shiller, 2017). We also find that newspaper text contains stronger signals of economic sentiment than economic uncertainty, although text-based measures of uncertainty have received greater attention to date.

In considering how to use text to understand and forecast key economic variables, we explore the pros and cons of several methods of turning text into time series in the context where that information would feed into policymakers’ judgements. Our most significant novel methodological contribution is the proposal and consideration of a feature engineering based approach that combines a large space of text-derived regressors with supervised linear and non-linear machine learning. We find that even simple counts of words perform surprisingly well but, of existing deterministic methods, a dictionary of words associated with financial stability offers the best all-round performance. However, the above mentioned machine learning approach materially outperforms all other methods – irrespective of the conditioning information set. Additionally, we find a plausible channel for text information to be relevant; variable importance exercises demonstrate that terms like 'price', 'market', 'growth', 'dire', 'taxes', 'house prices', and 'inflation' are picked up by our machine learning approach and drive its results. The approach we introduce is also likely to be applicable to forecasting problems in other contexts. Of the range of machine learning methods we compare, we find that (non-linear) neural networks consistently perform the best across text sources, time horizons, and target variables.

We make several key contributions relative to the existing literature. First, we forecast a wide range of macroeconomic variables using newspaper text. In doing so, we find that text can improve forecasts of many variables of interest to policymakers such as GDP, inflation, and unemployment. Forecast improvements are found at horizons of nine months into the future and especially occur during periods of stress. Second, we perform macroeconomic forecasting with text using rolling window re-estimation with the h-step ahead out-of-sample forecasts that are used in practice in policy[1]. In doing so, we provide one means for policymakers to derive significant value from text for macroeconomic forecasting. Third, we highlight the importance of text-based sentiment relative to text-based uncertainty. Fourth, we compare many different text-based measures from the literature in a horse race, including popular Boolean and dictionary- (or lexicon-)based methods. Fifth, we control for and comment on the many subtle pitfalls related to information leakage when making forecasts with text. Finally, we show that combining supervised machine learning and text-feature engineering can yield better forecasts than simpler methods for a wide range of target variables. Our feature engineering method creates a large number of regressors from text and turns each into a time series that is fed into a supervised machine learning algorithm. Demonstrating the forecasting success of this method, appropriate for high dimensional datasets, on a wide range of target variables is our main contribution – especially as it is transferable to other contexts where text could be used as an input into forecasts.

Several other papers have explored the link between text and economic activity (Gentzkow, Kelly and Taddy, 2017), for instance in the case of firms' annual reports and their returns (Jegadeesh and Wu, 2013; Loughran and McDonald, 2011, 2013), between newspaper text and levels of uncertainty (Alexopoulos and Cohen, 2015; Baker, Bloom and Davis, 2016), using survey responses (Borup et al., 2020b), and even in what people search for online (Borup and Schütte, 2020; D'Amuri and Marcucci,

[1] Results remain qualitatively similar when we apply an expanding window.

2017). A particularly relevant example linking text to uncertainty is that of Manela and Moreira (2017), who retrospectively forecast the VIX based on front-page articles in *The Wall Street Journal*. Their news implied volatility peaks in financial crises and rises just before transitions into economic disasters. There is other evidence that text is more strongly linked to financial market activity during periods of stress. Nyman et al. (2018) show that text-based measures of excitement rose substantially before the financial crisis and note that they may be an important warning sign of impending financial system distress. Garcia (2013) shows that news-derived sentiment can affect asset prices, and that the effect is particularly strong during recessions.

The closest paper to ours is Shapiro, Sudhof and Wilson (2018). They look at the ability of a number of dictionary (or lexicon) based sentiment analysis methods to predict the same 5 classifications (very negative to very positive) as human subjects on 800 newspaper articles. Unlike our paper, they do not perform out-of-sample forecasting tests with the sentiment series but do show that they co-move with the business cycle and correlate with survey-based sentiment measures. The news generated sentiment indices are used to estimate the impulse responses of macroeconomic variables to sentiment shocks. Shapiro, Sudhof and Wilson (2018) acknowledge that machine learning may have advantages but their training set is too small to be amenable to supervised machine learning. Here, we assess algorithms that do not learn (for instance, dictionary and Boolean methods) as well as supervised machine learning as our sample size is large enough for it to be effective in forecasting the (continuous) economic variables that most concern policymakers. More recently, Ellingsen, Larsen and Thorsrud (2021) has shown that news text data contains information not captured by hard economic indicators, including information that is particularly useful in forecasting consumption.

There is a growing literature on forecasting with text that began with forecasts of financial markets and firms (Antweiler and Frank, 2004; Tetlock, 2007). Our results looking at a range of macroeconomic target variables build on the findings of Thorsrud (2018) and Larsen and Thorsrud (2019), who use Norwegian newspaper text to predict output based on an unsupervised text approach, Latent Dirichlet Allocation (LDA)[2], and find that nowcasts using text are broadly competitive to those based on expert judgement or a model combination framework. Similarly, both Ardia, Bluteau and Boudt (2019), using US newspaper text, and Rambaccussing and Kwiatkowski (2020), using UK newspaper text, combine expert judgement and linear machine learning to forecast output. Our feature engineering approach

[2] LDA is a type of unsupervised learning known as topic modelling. While unsupervised machine learning looks for patterns within inputs, supervised machine learning is more analogous to regression: it looks for patterns between inputs and outputs.

is closest to the one proposed by Manela and Moreira (2017) and Kelly, Manela and Moreira (2019), however, our application differs in many respects. In particular, the former study focuses on financial forecasting using support vector regression while we opt to forecast macroeconomic fundamentals using a broad range of linear and non-linear supervised machine learning algorithms. In addition, Kelly, Manela and Moreira (2019) develop a new text selection methodology, the hurdle distributed multinomial regression (HDMR), while we aim to compare the performance of a set of distinct machine learning techniques using a new UK dataset. More recently, Bybee et al. (2020) has shown that time series derived from themes captured by topic models can provide incremental power in macroeconomic forecasts.

We focus on supervised machine learning models that can easily be re-estimated in an out-of-sample exercise that simulates forecasting in policy-making institutions. Even though results of now- and forecasting with topic models are promising (Thorsrud, 2018), the use of these models in a rolling forecast environment can be computationally infeasible and introduces an identification problem because there is no guarantee of any topic similarity following re-estimation. Therefore, we do not include topic modelling in the models that we use in this paper.

The rest of the paper is organised as follows: we first describe our newspaper text data in §2. We then discuss the different methods to turn text into time series in §3, beginning with our discussion of the pitfalls of using text data in real-time before describing algorithm-based text metrics in §3.1 followed by machine learning based measures in §3.2. In §4 we look at whether the algorithm-based text metrics can function as indicators by comparing them to a suite of existing indicators used by policymakers. §5 gives the background on both types of forecasting exercises that we perform before §5.1 uses algorithm-based text metrics in forecast exercises and §5.2 looks at the forecast performance of the machine learning based approach. §6 discusses the overall results and concludes.

2 Data

Our data are from Dow Jones Factiva and comprise three popular UK newspapers: *The Guardian*, *The Daily Mail*, and *The Daily Mirror*. Newspaper articles are retrieved through an application programming interface (API) which filters for the subjects Commodity/Financial Market News, Corporate/Industrial News, and Economic News. The allowed article types include regular news, editorials, and commentaries/opinions and these articles could be featured both online and in print. Regular news stories are written by journalists, while commentary and opinion pieces could be written by either a journalist or guest writer. We restrict to these types articles for two main reasons. The first is to maximise the signal-to-noise ratio; articles about other topics, such as sport, will still carry a sentiment but not one that is necessarily economically meaningful. The second is for computational feasibility.

We discard any articles that are updates of previous articles on the basis that the most salient information, if newsworthy, would have been in the first release. We also discard any articles with exactly the same text content as another article, keeping only the first occurrence of such articles, and removing any remaining duplicates through string matching. Such articles are not uncommon in this corpus (Eckley, 2015). Descriptive summary statistics of the newspapers are shown in Table 1, which shows the number of unique articles after de-duplication. The circulations shown in the table are for June 2018 (Newsworks, 2018).

Ofcom, the UK's communications regulator, estimates that *The Daily Mail* has a readership in which 37% of readers are over 65 but which is evenly split by socio-economic grouping, *The Guardian* has a readership in which only 4% of readers are over 65 and it contains a larger proportion of people in the ABC1 socio-economic group, while *The Daily Mirror* has a readership that is more evenly distributed by age and contains a larger proportion of people in the C2DE socio-economic group.[3]

Our motivation for the inclusion of these newspapers is twofold: each is available in a digital format back to the 1990s, giving a longer period to test forecast performance, and each has a significant national circulation over the period we study. Because these newspapers are widely read, they can potentially influence the decisions that households across the UK make; and indeed we find that, for example, *The Daily Mail* is particularly powerful at forecasting household consumption.

	Circulation	Unique articles	% of total	⟨articles/month⟩	First article	Last article
The Guardian	138,000	288,928	54.7	828	1990-01-06	2019-01-23
The Daily Mirror	563,000	141,332	26.8	492	1995-03-01	2019-01-23
The Daily Mail	1,265,000	97,897	18.5	281	1990-01-11	2019-01-23
Total	1,966,000	528,157	100.0	1,601	-	-

Table 1: Descriptive statistics of articles from selected UK newspapers. "% of total" refers to the percentage that a particular news source contributes to the total number of articles. Source: Dow Jones Factiva.

3 Turning text into time series

For the methods that we use, the text of each newspaper article must be pre-processed before being transformed into numbers. This step is also known as text cleaning. We remove punctuation and

[3] From the *Ofcom News Consumption Survey 2018.*

digits, enforce lower case, and remove a large number of stopwords – words that are not by themselves informative, typically conjunctions such as 'and'. We use two approaches to turn pre-processed text into quantitative time series that are then used as inputs into forecasts: algorithm-based text metrics[4] and term frequency vectors. Throughout, we refer to terms rather than words, as these are more flexible. A term could be composed of one word, two words, e.g. 'bank run'. This is known as a 2-gram, and a phrase of length N as an N-gram. Or a term could be the stem of a word, e.g. 'econom' for 'economics' and 'economy'. More details of text cleaning may be found in Appendix A.

We only include those methods from the existing literature that can be feasibly computed in real time, including being re-estimated at every time step. For this reason, we exclude topic models.[5]

Where necessary, we have also modified existing methods to exclude information about the future, to prevent information leakage, which is also known in this context as look-ahead bias.[6] The simplest example of information leakage with time series is when a continuous time series variable is normalised, i.e. $x_t \longrightarrow \frac{x_t - \mu_x}{\sigma_x}$ where the mean, μ_x, and standard deviation, σ_x, take the entire sample, $\{x_t\}_{t=0}^{t=T}$, as their domain. In real time, at time t, information on times $> t$ is not available and so the transform should be time-dependent, i.e. $x_t \longrightarrow \left(x_t - \mu_{\{x_{t'}\}_0^t}\right) / \left(\sigma_{\{x_{t'}\}_0^t}\right)$. This example may be trivial, but there are other ways for information leakage to occur with text. Just as typical time series can undergo global transforms that should account for time-dependent means and standard deviations, so too can text based transformations.

The most common text-based information leakage occurs during pre-processing of text. It is usually undesirable to track every single possible term or combination of terms in a corpus. Typically, a decision is made to omit certain words from analysis, for example those that occur very frequently or very infrequently in the corpus. This is usually done by specifying both a minimum and maximum threshold frequency that omits frequent but uninformative words, such as 'the', as well as words that are so rare as to be statistically irrelevant. But threshold frequencies assume and require knowledge of all words in the corpus, which is not possible in real-time. Terms that suddenly appear at one point in time can be correlated with macroeconomic developments but may only be tracked because they began to appear at a certain point in time. A good example would be the term 'sub-prime': this might pass

[4] These are algorithms with a fixed relationship between input and output that do not involve any learning.

[5] As Thorsrud (2018) points out, recursive updating of the topic model is computationally expensive and has an identification problem: even in dynamic topic models (Blei and Lafferty, 2006), the same topics cannot be guaranteed to appear, or to be linked, when the model is re-estimated.

[6] One potential source of data leakage that we cannot address here is that the Dow Jones Factiva subject filter that we use to select articles may have changed over time to reflect events that have occurred. While such concerns are valid, we have no easy way to avoid them with this data source.

a whole-corpus threshold filter but would be far less likely to pass the same filter applied only to text from before 2007. So tracking such a word might appear to produce very strong results that would not have been possible in real time. Such issues can apply to dictionary, Boolean, topic, and machine learning models alike. We explain how we avoid this in §3.2.

Finally, when using text in the context of machine learning it is advisable to consider whether and how the training and test sets may differ. In our case, both are drawn from the same newspaper sources but we mention it here as it could be an issue when using pre-trained models such as BERT (Devlin et al., 2018).

3.1 Algorithm-based text metrics

We define algorithm-based text metrics as pre-defined rules, or algorithms, that turn text into numbers without any element of learning. They are by far the most commonly used method to extract information from text. The simplest example that we use in this paper is the count of the number of times a specific term appears in each article divided by the number of words in the article. The numerical scores for a particular month are found from the mean of the scores of the articles that were published in that month.

The set of algorithms we use to create text metrics is summarised in Table 2. They fall into three broad categories (see Appendix B for formal definitions of each).

Dictionary methods typically associate specific terms with specific scores (positive or negative for sentiment) and count the net score per article. The dictionaries that we include cover financial stability (Correa et al., 2017), finance (Loughran and McDonald, 2013), social media sentiment from Nielsen (2011) and named here as 'Afinn', psychological terms (from the Harvard IV psychological dictionary as used in Tetlock (2007)), and common English words that measure a score between the emotions of anxiety and excitement (Nyman et al., 2018).

We also use variations on dictionary measures that are even simpler: word counts (also known as term frequencies, here weighted by article lengths), and transformed word counts. We use the single term counts of "uncertain" and "econom". We also use a more sophisticated weighting, the term frequency – inverse document frequency (tf-idf). This seeks to control for the frequency of the term in each article ($\text{tf}(a)_w$), the number of articles per day (N_t), and the number of articles in which the term appears per day ($n_t < N_t$). We use the number of articles in which the term appears each day in place of the usual number of articles in which the term appears through the whole corpus to avoid information leakage. This measure uses a log transform, partly mindful of the power law for the

Positive and negative dictionary	Boolean	Computer science-based
Financial stability (Correa et al., 2017)	Economic Uncertainty (Alexopoulos and Cohen, 2009)	VADER sentiment (Gilbert, 2014)
Finance oriented (Loughran and McDonald, 2013)	Monetary policy uncertainty (Husted, Rogers and Sun, 2017)	'Opinion' sentiment (Hu et al., 2017; Hu and Liu, 2004)
Afinn sentiment (Nielsen, 2011)	Economic Policy Uncertainty (Baker, Bloom and Davis, 2016)	
Harvard IV (used in Tetlock (2007))		
Anxiety-excitement (Nyman et al., 2018)		
Single word counts of "uncertain" and "econom"		
tf-idf applied to "uncertain" and "econom"		

Table 2: The three broad categories of algorithm-based text metrics used.

frequency of different terms in the English language (Zipf, 1950):

$$\text{tf-idf}(a)_t = \frac{\ln\left(1 + \text{tf}(a)_w\right)}{\ln\left(1 + N_t/n_t\right)}$$

Boolean methods provide a count of articles only if the terms in an article satisfy some logical condition, for instance that three distinct and pre-defined terms all appear in the same article. In the most simple case, this just counts any article that contains a specific term. The most notable examples of Boolean methods are the Economic Uncertainty index of Alexopoulos and Cohen (2009) and the similar UK version of the Economic Policy Uncertainty (EPU) index of Baker, Bloom and Davis (2016). However, note that while we apply the text analysis methodology of the UK EPU index, Baker, Bloom and Davis (2016)'s paper uses *The Times* and *The Financial Times*, different publications to ours, and they include all articles, not just those about economic developments.[7]

Our third type of metric draws on the computer science literature. Two of the metrics that we implement are from previous research; the VADER metric (Gilbert, 2014) is rule-based and designed for sentiment as expressed on social media while the opinion sentiment metric (Hu et al., 2017; Hu and Liu, 2004) combines machine learning and product reviews to develop a dictionary-based method.

Table 3 shows the scores produced by some of the algorithms for example articles. We add prefactors of -1 to some metrics to ensure that both positive sentiment and heightened uncertainty receive a score that's greater than zero. The consistency of sign will be useful in subsequent plots. Negative signs before zero indicate that the scores were more than -0.01 but less than zero.

There are examples from each of the three types of metric shown in Table 2. The first piece of text is taken from the February 2018 Bank of England *Inflation Report* and, according to the metrics, is positive in sentiment.[8] The second is fictional and designed to encapsulate high uncertainty and

[7] Note that the real-time EPU UK time series available on their website uses a different combination of 11 UK newspapers.

[8] In the interest of space only part of the text is shown in the table.

Text	TFIDF economy	Vader	Counts economy	Alexopoulos	Stability
Global GDP growth picked up during 2016 and has been strong over the past year (Section 1.1). Weighted by countries' shares of UK exports, global growth is estimated to have remained at 0.8% in 2017 Q4. That pace of growth is expected to persist in the near term, above expectations in November. Survey indicators of output (Chart 1.1) and new orders remain robust, particularly in the euro area and United States. Measures of business and consumer confidence are also healthy...	-0.00	0.97	0	0	0.03
The economy has struggled and is in a bad state with disappointing performance, unhappy consumers, low confidence with high uncertainty. Policy faces a number of risks which could transmit to the real economy, and pundits are increasingly concerned about a crash.	-0.15	-0.93	-2	1	-0.11
The current direction of policy is very bad.	-0.00	-0.54	0	0	-0.25
The current direction of policy is very good.	-0.00	0.44	0	0	0.25

Table 3: Selected algorithm-based text metrics applied to example text. In the interest of space, the first text example is truncated. For sentiment, both magnitude and sign matter. For uncertainty, only magnitude is relevant. We give pre-factors of -1 to some metrics so that positive sentiment has a positive score, for example, the 'Counts economy' score is defined to be -1 times the number of counts of the word economy, on the basis that discussions of the economy are likely to be due to negative sentiment about the economy. Negative signs before zero indicate that the scores were more than -0.01 but less than zero. Heightened uncertainty, for example in the Alexopoulos and Cohen (2009) measure, has a positive score.

negative sentiment. Note that only the second text entry triggers the Boolean Alexopoulos metric, because that text contains the word 'uncertainty' and 'economy'. The third and fourth text examples are very similar, but with 'bad' replaced by its antonym, 'good', and, in consequence, almost reversed sentiment scores.

3.2 Machine learning methods

We now describe our alternative method of employing text for economic forecasting. This method does not create time series that function as indicators, unlike the algorithm-based text metrics, but is well-suited to forecasting with text. In particular, it seeks both to extract as much of the rich information available in the text as possible and to allow the model to decide which terms to put weight on in real-time, rather than fixing this ahead of time. We achieve this in two steps, exploiting a combination of feature engineering and supervised machine learning.

The former step creates a large set of features (in the language of machine learning) or regressors (in the language of econometrics) to use as the inputs to a machine learning algorithm that can operate with a greater number of features than observations. This large feature space allows for a broader set of the information in the text to be captured. The feature engineering that we choose represents each article as a term frequency vector. Term frequency vectors extend the idea of counting terms to a large number of terms.

The term frequency for a term w in an article a is denoted $\text{tf}(a)_w$ and is simply the counts of term w in that article. Term frequency vectors are the vector representation of all (tracked) terms in an article or across articles in a given time period t. For example, for articles, the term frequencies define a vector space: $V : a \to \mathbb{R}^N$ with N the dimension of the vector space and, equivalently, the number of tracked terms. A complete matrix, tf, may also be defined in which each column is a term from the pre-defined set of all terms, and each row is an observation (an article or collection of articles within a time period).

We use 9660 terms, with up to 3-grams. The pre-defined list of terms used to construct the term frequency matrix uses the union of several dictionaries. These are those dictionaries found in Nyman et al. (2018), Loughran and McDonald (2013), Nielsen (2011), Hu and Liu (2004) and Hu et al. (2017), and Correa et al. (2017). We add to this a collection of words related to economics and finance[9] and the Harvard IV psychological dictionary used by Tetlock (2007). We use n-grams up to trigrams only if they already exist individually in these dictionaries. For example, "interest rate risk" is a tri-gram inherited from one of the component dictionaries. This gives 9660 unique terms of which 8030 appear in our corpus.

It is because the dictionaries that we use are drawn from other studies and are independent of our corpus that some of the terms never appear in our corpus. To use term frequency vectors as inputs into forecasts, each article is represented as a vector (one dimension for each term) of counts of terms that occur within it. Each vector may be denoted

$$\overrightarrow{\text{tf}(a)} = (\text{tf}(a)_{w_1}, \text{tf}(a)_{w_2}, \cdots)$$

The term frequency vector for a month is the mean of the vectors of the articles published in that month.

In the second step, we use supervised machine learning models to automatically decide which of this large set of terms (or combinations of terms) to put weight on by using the term frequency vectors as features (regressors). This approach contrasts with forecasts using dictionary-based text metrics in which a pre-determined set of weights are effectively applied to terms to create a net score. That

[9] Most of these come from https://home.ubalt.edu/ntsbarsh/stat-data/KeywordsPhra.htm and http://home.ubalt.edu/ntsbarsh/Business-stat/stat-data/KeysPhrasFinance.htm.

aggregate net score is then used as a regressor in forecasts. Instead, in this approach, the weights on individual terms are set by the supervised machine learning model, a more flexible solution. In general, this is likely to produce better predictions than specifying some of the weights in advance as the method effectively searches over a wider space of possible terms and weights.

The machine learning stage uses the term frequency vectors to predict a target variable y at time $t+h$:

$$\hat{y}_{t+h} = f_{\text{ML}}\left(\ldots, \overrightarrow{\text{tf}}_t\right)$$

where f_{ML} is the function obtained through training a machine learning model and $\hat{y}$ is a forecast of y. We use a number of machine learning algorithms detailed in Appendix E; lasso regression, ridge regression, elastic net regression, support vector machine regression (SVM), random forest, and a shallow artificial neural network (NN).

4 Algorithm-based text metrics as proxies

We now turn to our first set of results and ask whether algorithm-based text metrics can be used as plausible forward looking indicators and, if so, which are the most effective?

In answering this question we break down algorithm-based text metrics into two varieties: those which track sentiment, and which we would expect to track economic growth, and those which track uncertainty, and which we would expect to track more traditional measures of uncertainty. This is a convenient split because these are two quite different concepts, because measures of sentiment and uncertainty already exist in the text analysis literature, and because measures of sentiment and uncertainty have direct analogs among traditional economic measures. These are also *useful* concepts: measures of sentiment are useful to policymakers because they are well-correlated with, or act as leading indicators of, many different measures of realised economic activity. They may also capture Keynes' *animal spirits* and Shiller's idea (Shiller, 2017) of narratives that spread like viruses. Likewise, measures of uncertainty are useful for policymakers precisely because they are difficult to measure directly through real activity though they do have effects on economic activity, for instance in delaying the consumption of durable goods. The y-axes of Figures 2 and 4 show which text-based metric is assigned to sentiment and uncertainty respectively, and more details of all algorithm-based text metrics may be found in Table 2.

First, though, we run an augmented Dickey-Fuller test for stationarity on all text-based time series, finding that almost all of our series are stationary (the results are in Appendix B.2). The small number of newspaper-series pairs that aren't tend to be based on counts of a single term or use boolean article counting and are most commonly found with the text of *The Guardian* newspaper.

We separate the rest of our analysis of text-based time series into measures that track either sentiment or uncertainty. The non-text "proxies" we use are traditional macroeconomic and financial indicators, and are chosen as being representative of the indicators policymakers might currently use. To these we also add recently developed series focusing on uncertainty from the academic literature. Note that the investment grade corporate bond spread could be considered to contain signals of both uncertainty and sentiment (with, one would expect, opposite signs), and so we include it in both correlation heatmaps. Full descriptions of all the proxies may be found in Table 4.

We compare each text-based time series to existing time series that proxy for sentiment and uncertainty both through visual inspection and by looking at the correlation of each text metric (averaged across the three newspapers) with the proxies 3 months ahead.

To visually compare the text-based metrics and the proxies, we use the average of all text metrics over time plotted against a swathe from the existing numerical proxies from Table 4. All text series are aggregated to monthly frequency using a 3 month rolling mean. In the interest of space, we show only two example plots that have particular features of interest: *The Daily Mail* for sentiment and for *The Guardian* for uncertainty.

Name	Description	Proxy for	Type
Lloyds Bus Conf	Lloyds Business Barometer – confidence	Sentiment	Survey
Lloyds Bus Activity	Lloyds Business Barometer – activity over next 12 months	Sentiment	Survey
OECD Bus Conf	OECD UK business confidence	Sentiment	Survey
Composite PMI	Composite measure of PMI	Sentiment	Survey
GfK Consumer Conf	GfK Consumer Confidence	Sentiment	Survey
IG Corp Bond spread	Investment Grade Corporate Bond spread	Uncertainty, sentiment	High-frequency market-based
Jurardo Fin Uncert	UK version of Jurado, Ludvigson and Ng (2015) from Redl (2018); financial uncertainty, $h = 3$	Uncertainty	Forecast error
Jurardo Macro Uncert	UK version of Jurado, Ludvigson and Ng (2015) from Redl (2018); macroeconomic uncertainty, $h = 3$	Uncertainty	Forecast error
BoE agg credit spread	Bank of England measure of aggregate credit spread	Uncertainty	Market-based
VIX	CBOE volatility index	Uncertainty	High-frequency market-based
VFTSEIX	FTSE volatility	Uncertainty	High-frequency market-based
GDP forecast std dev	UK Treasury collected standard deviation of professional forecasts of GDP, 3 months ahead	Uncertainty	Low-frequency forecast spread
BoE Uncert	Bank of England uncertainty index	Uncertainty	Composite
ERI volatility	GBP Exchange Rate Index volatility	Uncertainty	High-frequency market-based

Table 4: Descriptions of the "proxy" time series that are used as indicators of sentiment and/or uncertainty by policymakers. We compare algorithm based text metrics to these proxies.

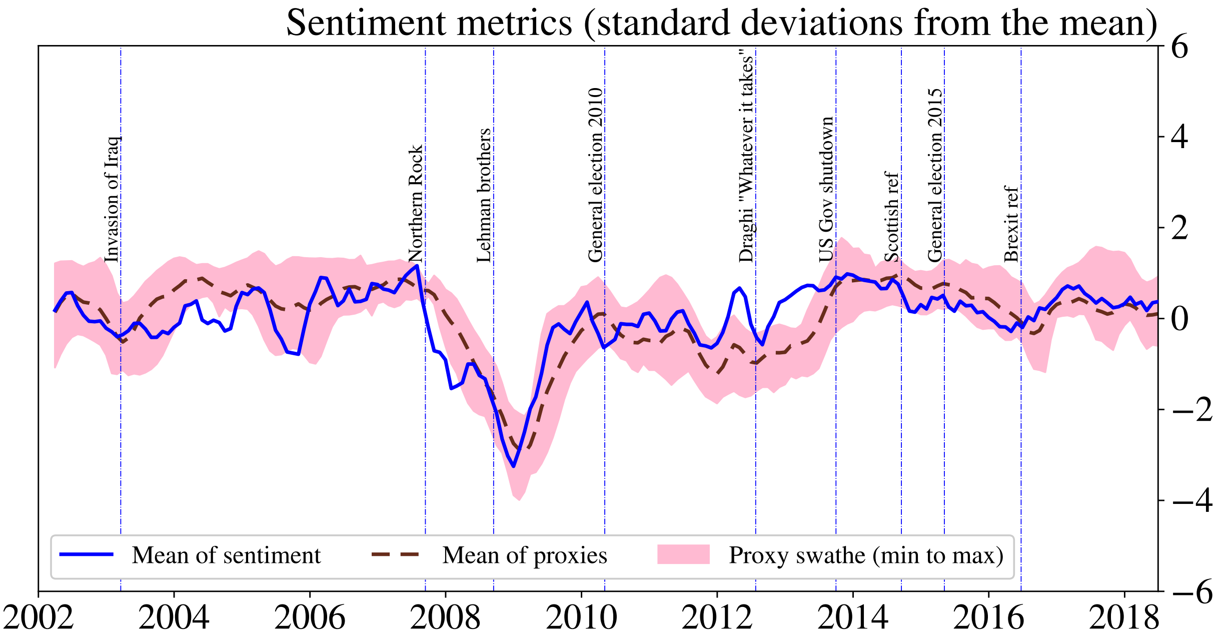

Figure 1: Three month rolling mean of the macroeconomic sentiment text metrics created from the text of *The Daily Mail* (solid line) plotted against the three month rolling mean of the proxies for macroeconomic sentiment (broken line) and a swathe defined by the maximum and minimum values across proxies at each point in time. The text-based sentiment measure tracks the usual proxies for sentiment well.

4.1 Sentiment

Fig. 1 shows the mean of all text-based series for sentiment versus a mean dotted line and swathe (min to max) for the proxies for sentiment. Fig. 1 shows that this broad measure of sentiment taken by averaging the text metrics has a striking qualitative correlation to the swathe of proxies. Of particular note is the sharp deterioration of sentiment that slightly leads, and then tracks, the global financial crisis. The leading nature of the text-based sentiment proxy is seen during the recovery too. There are periods when the sentiment indicator diverges from the mean of other indicators substantially, typically leading it. Note that although we have articles for *The Daily Mail* back to 1990, series as far back as this were not available for many of the proxy indicators, so, to ensure that the mean proxy measure is consistent over time, we restrict to the period from 2002 onwards when all the proxies we employ are available (the Lloyds Business Barometer indices are the limiting factors).

The correlation heatmap for sentiment is shown in Fig. 2. The correlations between the text metrics and the business confidence measure of the OECD are highest, and the most highly correlated text metrics are Stability and TFIDF (term frequency - inverse document frequency) economy. In general, the correlation between the text metrics and the proxies is appreciable and of the expected sign, but there are also a number of weak correlations. The correlations use the values of the traditional indicators three months ahead of the text indicators; the pattern of correlations persists at 6 and 9 months but become weaker as the horizon is increased.

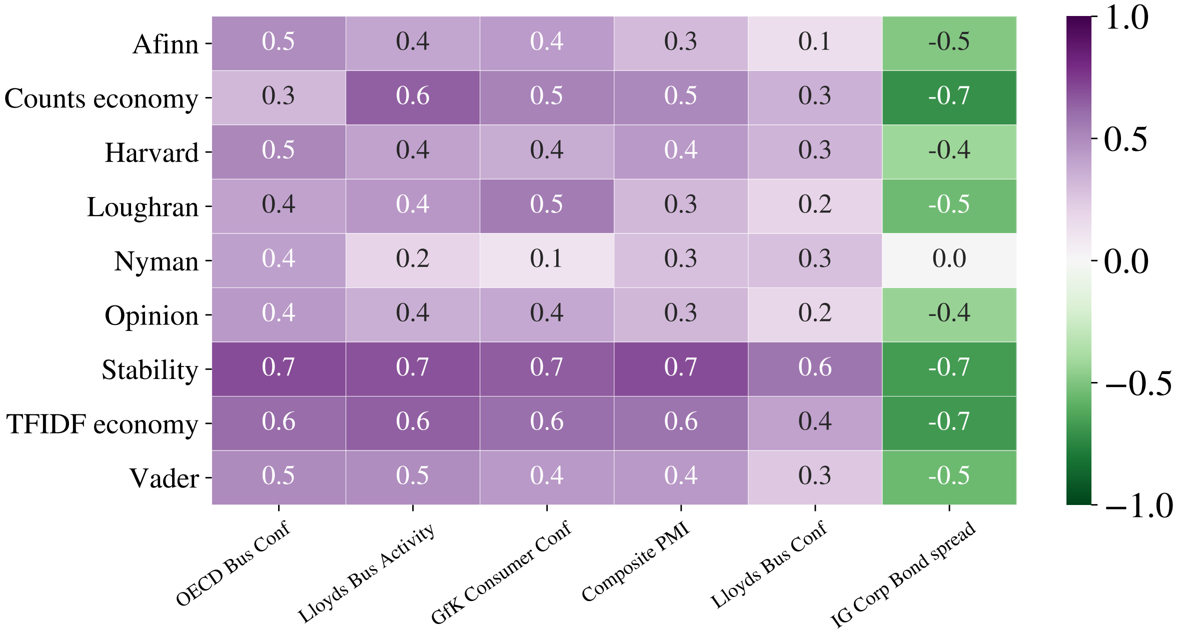

Figure 2: Heatmap of correlations between text metrics (averaged over newspapers) and proxies for macroeconomic sentiment at a three month horizon. Full definitions of the proxies may be found in Table 4.

4.2 Uncertainty

Fig. 3, showing uncertainty, does not reflect uncertainty proxies as well as was the case for sentiment, especially during the global financial crisis and around the Brexit referendum. Overall, the uncertainty measures based on text put more weight on events that are UK-specific. For example, they respond more strongly to the invasion of Iraq, the run on Northern Rock, and public votes within the UK, especially the Brexit referendum. Part of the difference could be because the newspapers we analyse are strongly UK-focused and do not tend to report events in financial markets, whereas our proxies for uncertainty include non-UK specific measures such as the VIX. For consistency with the sentiment series, the uncertainty index based on text is shown from 2002 onwards.

The lack of a strong increase in uncertainty during the global financial crisis is consistent with other text based measures of uncertainty, such as the EPU-UK index of Baker, Bloom and Davis (2016) (even

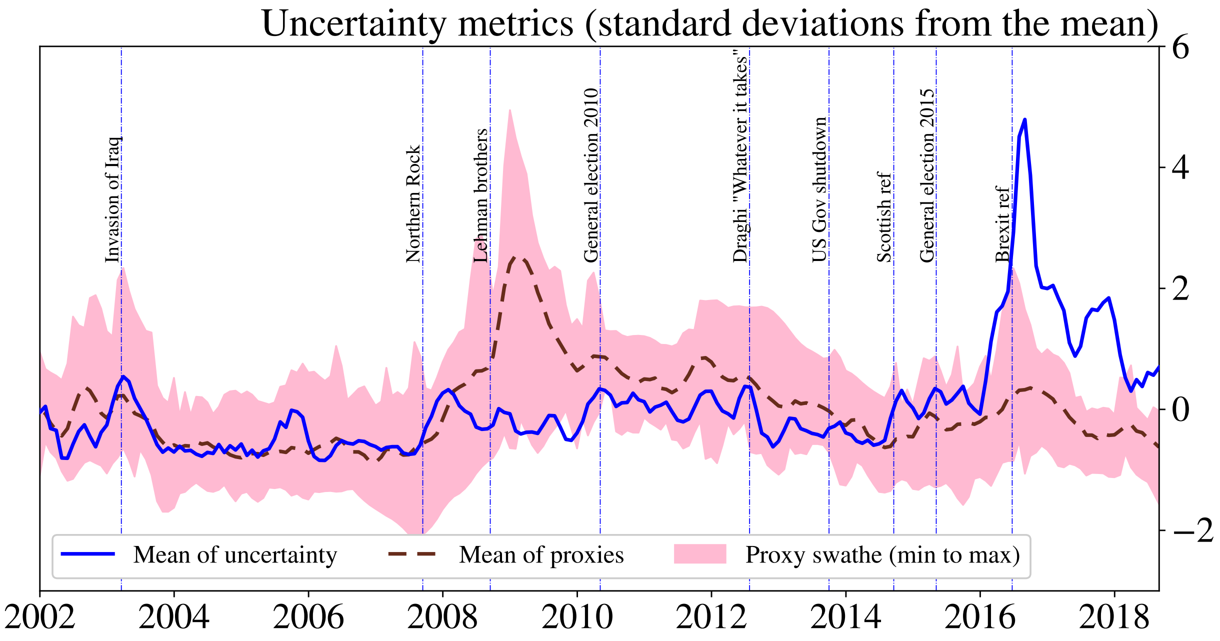

Figure 3: Three month rolling mean of the macroeconomic uncertainty text metrics created from the text of *The Guardian* (solid line) plotted against the three month rolling mean of the proxies for macroeconomic uncertainty (broken line) and a swathe defined by the maximum and minimum values across proxies at each point in time. The very large increase in uncertainty towards the end of the series coincides with the UK's referendum on whether to leave the European Union.

though it uses a different set of newspapers) and the non-Euro area uncertainty index of Mumtaz and Musso (2018).

The correlation heatmap for uncertainty is in Fig. 4. Generally, there is consistent weak correlation between the text based measures of uncertainty and the proxies for uncertainty. With the exception of the VIX and VFTSEIX, and putting aside the weak performing Husted index, the text based measures are more correlated with the faster moving proxies – ERI volatility, corporate bond spreads, and aggregate credit spread – than the slower moving measures like the standard deviation of GDP forecasts. The correlations use the values of the traditional indicators three months ahead of the text indicators; the pattern of correlations persists at 6 and 9 months but become weaker as the horizon is increased. The correlations for sentiment are substantially stronger than shown here for uncertainty. A similar result is shown by Kozeniauskas, Orlik and Veldkamp (2018) who document the weak correlations across a wide range of uncertainty proxies used in the literature.

For uncertainty, the measure using the method from Alexopoulos and Cohen (2009) and the similar measure from Baker, Bloom and Davis (2016) are also highly correlated. The measure of Husted, Rogers

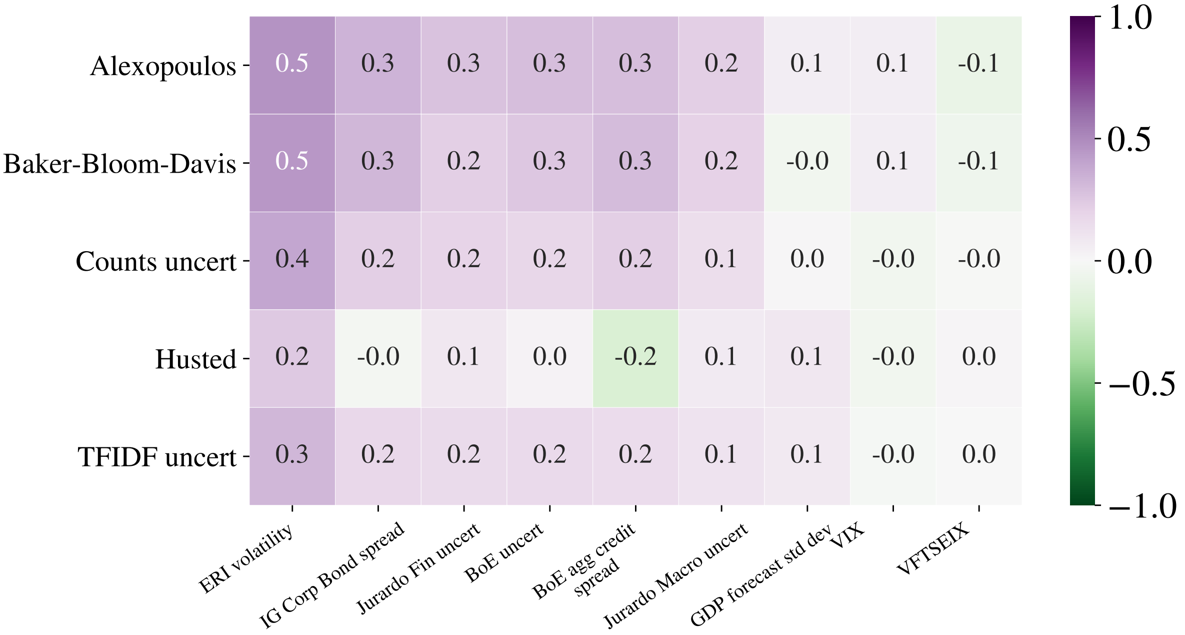

Figure 4: Heatmap of correlations between text metrics (averaged over newspapers) and proxies for uncertainty at a three month horizon. Full definitions of the proxies may be found in Table 4.

and Sun (2017) measures monetary policy uncertainty specifically and this is likely behind its lower levels of correlation with the more general uncertainty metrics. This suggests that counts of the word uncertainty are providing most of the power of the indicator.

4.3 Granger Causality

We also ask whether our sentiment or uncertainty text metrics Granger cause any of their relevant proxies and vice versa. The results are in Tables B.3 and B.4 of Appendix B.3.

For the sentiment metrics, the Stability metric, counts of the stem of the word economy, and TFIDF economy all Granger cause a large number of proxies. The Stability metric, from Correa et al. (2017), is the strongest performer overall as it Granger causes a large number of proxies at the 1% significance level, is stationary, and has the highest average correlations with proxies. This is unexpected as the Stability metric is a dictionary designed for a financial stability context, specifically the *Financial Stability Reports* of many countries' central banks rather than for the text of newspapers aimed at the general public. Yet, many of its words could plausibly be used to describe the economy in newspapers, for instance 'rebounding', 'sluggishness', and 'over-heated'.

Table B.3 shows that the uncertainty metrics tend to Granger cause the EPU UK index and the

UK version of the macroeconomic uncertainty indicator of Jurado, Ludvigson and Ng (2015) from Redl (2018), but not the equivalent financial uncertainty indicator. The simplest uncertainty metric, counting the stem of the word uncertainty, also Granger causes the investment grade corporate bond spread. In general, the performance of macroeconomic or financial uncertainty is weaker, and much more mixed across the analysis.

4.4 Summary

Taken collectively, this section highlights that text metrics can and do capture, in a forward-looking and timely way, some of the same information as the proxies that policymakers typically look at. Text metrics for macroeconomic sentiment show the strongest relationship with existing proxies and this is likely to be due to the nature of the news sources. Across these tests, the text metrics that perform consistently well are TFIDF economy and Stability for sentiment, with a more mixed picture for uncertainty.

One caveat to these conclusions is that the Stability metric is designed to capture financial stress so its good performance may reflect that it either tracks terms that its creators could only have known about with hindsight or because a substantial amount of our variation occurs over the financial crisis, for which this metric is naturally suited. The former risk is low as the dictionary behind the Stability metric contains no proper nouns and almost all of its words are general, e.g. ‘sluggish’, with only a small number of specialist financial words, e.g. ‘write-downs’, and no words that would specifically and solely tie it to the global financial crisis. The latter risk is more material but, as we will see in the next section, our preferred way of obtaining information from text will not rely on any pre-constructed text metric.

5 Forecasting exercises

Forecast exercises involve estimating a model over a limited training period followed by out-of-sample predictions of target variables at given horizons. Models are re-estimated at every step in time according to a 36 + 1 month rolling window (details of our rolling window forecasting environment are available in Appendix C.2). If any transforms of the features are carried out, for instance normalisation, they are only performed with data from the past or present. We use each newspaper separately to perform forecasts, rather than pooling them. While a pooled approach would have been equally valid, keeping the newspapers separate means that we can construct standard deviations for forecast performance and so get a sense of the generalisability of the performance of the text-based forecast methods that we examine. A detailed description of the training and testing exercise we run may be found in Appendix C.

Our forecast exercise seeks to answer whether a model with text included outperforms a very similar model with text not included. To reflect the timeliness of text, each forecast is done as if a policymaker at time t has information from text at time t, given by a scalar or vector indexed by time x_t, but information on the target variable y from time $t-1$, y_{t-1}, and before only. The policymaker wishes to forecast what y will be h steps in the future, y_{t+h}. This time t scenario of having potentially stale information on y but having access to newspaper text is of great relevance to policy where many official statistical series and survey data only appear with a lag. The baseline model without text that we use for comparison is either an AR(1) or a factor model (that combines many relevant time series) regardless of the target of the forecasting exercise. Although AR(1) models are simple, there is overwhelming evidence that, on average, and across series and time periods, they are tough to beat (Carriero, Galvão and Kapetanios, 2018).

Note that some of the hard indicators and targets we use in models as a lagged variable at one timestep (i.e. at y_{t-1}) would *not* be available to plug into forecasts in real-time because they are only released with a lag. However, we choose to use this data regardless for two reasons. The first is to keep our forecast exercises more consistent, simple, and transparent: including different release schedules would add substantial complexity to the analysis, especially given the nature of revisions to official data in the UK. The second is that if the addition of text can outperform a benchmark that has a somewhat unfair boost from using data that would not yet have been released, it is extremely likely that it would be even more performative versus a real-time benchmark that *does* take the release schedule of the target into account. We wish to err on the side of caution when examining the value that text can add; setting a tough benchmark is part of this.

Note also that while the plots in previous sections used a three-month rolling mean of text measures for visualisation purposes, the forecasts only use text data aggregated to monthly frequency.

Our forecasting targets are GDP, the unemployment rate, business investment, household consumption, consumer price inflation (CPI), the index of production (IOP), the index of services (IOS), the financial stress index of Chatterjee et al. (2017), and the IMF financial conditions index for the UK. All variables are at monthly frequency, with the exception of investment and consumption, which are quarterly, and are up-sampled using interpolation through time from in-sample data points only. More details may be found in Table 5.

Note that this is only a pseudo real-time forecasting exercise for several reasons. First, some of these series are revised, which we ignore here but this is not expected to favour text. Another is that the UK's monthly GDP index only began in 2018 and so was not available for some of the timeframe under consideration (it was, however, backdated to enable analyses of the kind we present).

Name	Description	Frequency	Transform
GDP	Gross Value Added: CVA SA	M	3M-on-3M growth
Fin stab index	Chatterjee et al. (2017) Financial Stress Indicator	M	None
CPI	CPI all items	M	Y-on-Y growth
IOP	Industrial Production	M	Y-on-Y growth
IMF fin cond	IMF UK Financial Condition Index	M	None
IOS	Index of Services	M	Y-on-Y growth
Unemployment	LFS Unemployment Rate	M	None
Hhld Consumption	Household Consumption	Q, up-sampled to M	Y-on-Y growth
Business Investment	Business Investment	Q, up-sampled to M	Y-on-Y growth

Table 5: Target variables for forecasts.

We forecast at horizons of $h = 3, 6, 9$ months; more details may be found in Appendix C.2. In the charts in §5.1 and §5.2, we plot error bars as the standard deviation of the forecast performance across both horizons and the different newspapers. Better forecast performance across horizons and newspapers is more indicative that the forecast gains are generally reliable and robust.

We now turn to the two types of regressor (and specification) that we use in forecasts: algorithm-based text metrics and term frequency vectors.

5.1 Forecasting with algorithm-based text metrics

We first evaluate the forecasting power of each text metric in turn using the model

$$y_{t+h} = \alpha + \beta \cdot y_{t-1} + \eta \cdot x_t + \epsilon_t$$

We compare the performance of this model to the same one without the text information present in x_t (i.e. we force $\eta \equiv 0$). Figure 5 shows out-of-sample forecast RMSEs (root mean squared errors) relative to our AR(1) baseline by metric and target variable.

There are strong improvements in the forecasts of GDP and its components compared to the baseline. Excluding CPI, forecasts of macroeconomic variables are improved by the addition of text while forecasts of financial conditions are little improved. Further statistics for the AR(1) benchmark may be found in Appendix D.1.

We now test text in a model that includes additional macroeconomic information. We utilise the macroeconomic factors derived from a dataset comprising 33 series covering real output, international trade, the labour market, inflation, house prices, retail sales, capacity utilisation, and business and

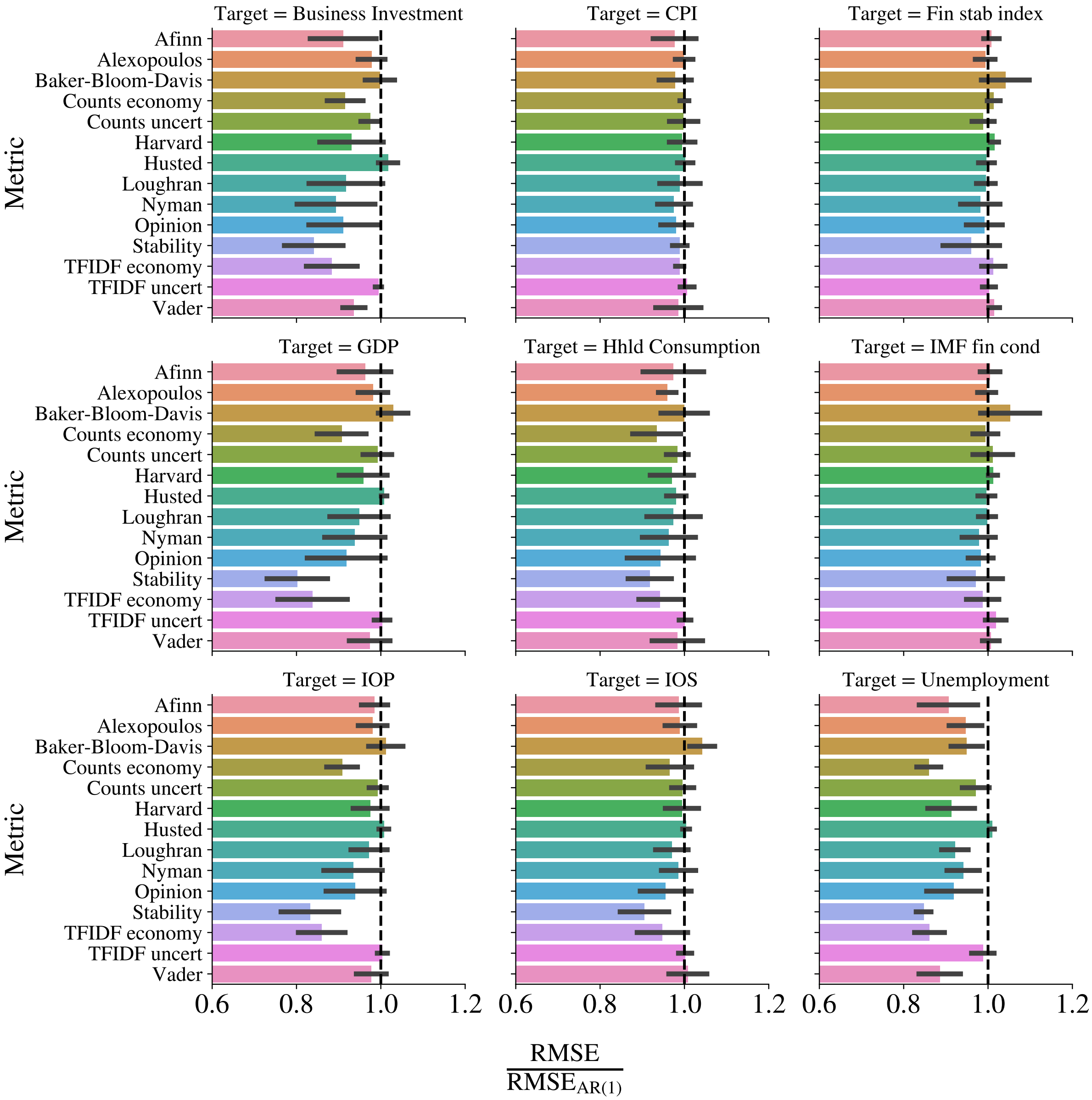

Figure 5: Results from the forecast exercise using algorithm-based text metrics. The plot shows RMSEs (x-axis) of a forecasting model with text in versus a benchmark AR(1) forecast without text. Facets are different target variables, the y-axis shows different algorithmic text metrics. Bars to the left of the dashed line indicate an improvement in forecast performance conducted with the given text metric. The confidence intervals are standard deviations over both the newspapers and the different horizons (3, 6 and 9 months ahead).

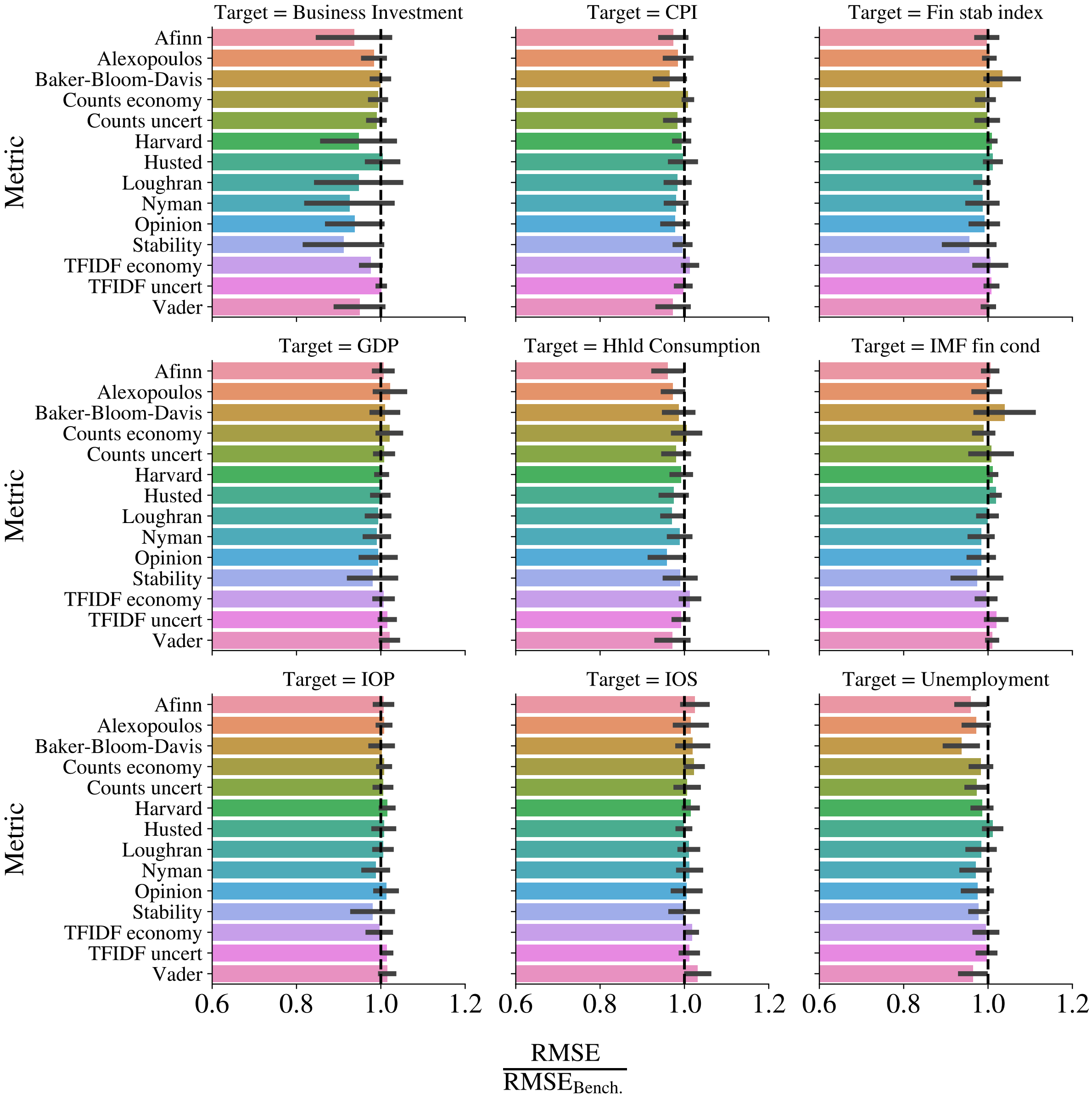

Figure 6: RMSEs relative to a benchmark AR(1) with factors by text metric and target variable. Bars to the left of the dashed line indicate an improvement in forecast performance conducted with the given text metric. The confidence intervals are standard deviations over both the newspapers and the different horizons (3, 6 and 9 months ahead).

household expectations (Redl, 2017)[10]. More information on the series is provided in Appendix I. The series used in the factors represent substantially more information than the proxies used in the correlational and visual evidence presented in previous sections. The factors are denoted by F. As before, the text model also includes a single algorithm-based text metric and an autoregressive term. The model is given by

$$y_{t+h} = \alpha + \beta \cdot y_{t-1} + \sum_j \gamma_j \cdot F_{jt} + \eta \cdot x_t + \epsilon_t$$

where x is the text metric and we use $J = 2$ factors as selected by the Bai and Ng (2002) statistic. The benchmark against which this is compared is the same model as above but without the term in x_t.

Figure 6 shows the forecast performance of the text and factors model. We find that the added value of text degrades significantly when the benchmark is changed to a richer factor model that has highly statistically significant confounders. Across all targets, the results are weaker than in the case of just using an AR(1) as a benchmark. Most notably, very few target-metric pairs offer forecast improvements across all horizons and newspapers. Some of the text metrics that perform well against the AR(1) benchmark retain their position in the rankings, such as the Stability metric, while others, such as TFIDF Economy, rank far worse. In general, the simple and transformed counts of single terms do not add as much forecasting power when compared with a tougher forecasting benchmark of a factor model (which is already fairly well suited to capturing general macroeconomic trends). Those metrics associated with financial markets and finance (Stability, Loughran, Nyman) seem to perform relatively better with this benchmark, perhaps reflecting that our factors are based on time series that mostly capture information on the real economy. Further statistics for the factor model benchmark may be found in Appendix D.2.

5.2 Forecasting with text and machine learning

Here we use a novel combination of term frequency vectors from newspaper text and supervised machine learning, with its ability to handle a large feature space, to make forecasts. We look at two cases, just as with algorithmic text metrics: forecasting versus a simple AR(1) model that uses OLS for estimation, and forecasting versus a richer factor model.

The models we employ from the machine learning literature are the least absolute shrinkage and selection operator (lasso) (Tibshirani, 1996), ridge regression (Hoerl and Kennard, 1970), support vector regression (svm) (Chang, 2011; Drucker et al., 1997), elastic net (Zou and Hastie, 2005), shallow artificial

[10] Note that these data are not real-time and so have a stronger information advantage relative to our text indices than real-time data would, ie they are likely to be more difficult to beat in a forecast than real-time data would be.

neural networks(Rumelhart, Hinton and Williams, 1985), and random forest (Breiman, 2001).[11] The exact specification of each is defined in Appendix E.[12]

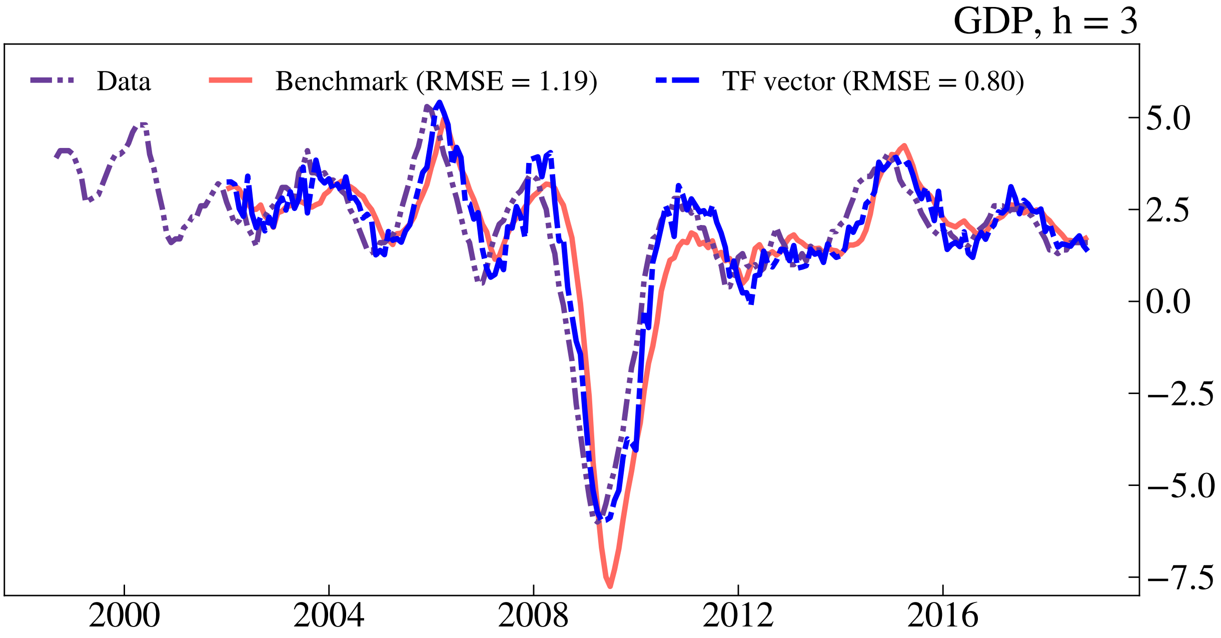

Figure 7: Forecasts for GDP growth three months ahead using OLS with a single lag (benchmark model, solid line) versus a shallow artificial neural network that uses term frequency vectors from newspaper text in addition to a single lag of GDP (TF vector, dash line). The data are also shown (dot-dash line). Both models are estimated using a rolling window and newspaper text is taken from *The Daily Mail*.

Let X represent the full $N \times T$ matrix of text-based features (counts of words in our custom dictionary) and $\vec{x}_t$ the same set of features at time t. We evaluate the forecasting power of each supervised machine learning model with text in turn using

$$\hat{y}_{t+h} = f\left(y_{t-1}, \vec{x}_t\right)$$

and we compare the performance of this model to the equivalent OLS model without the term, x_t, which includes text. We refer to this OLS only benchmark, given by $y_{t+h} = \alpha + \beta \cdot y_{t-1} + \epsilon_t$, as the OLS-AR(1) model. It is natural to ask why we do not compare the model with text against the

[11] Note that we do not use factor models on text, though factor models are closely related to ridge regression (Hastie, Tibshirani and Friedman, 2009).

[12] Note that we do not perform hyper-parameter tuning: running out-of-sample forecasting exercises with all of the possible combinations of newspapers, algorithms, and targets, is already extremely computationally intensive and tuning would have increased the dimensionality even further. We instead opt for fixed hyper-parameters. This does not affect our main results – that machine learning and feature engineering together can produce marginal improvements – because neither our preferred benchmark model nor the algorithm-based text metrics have hyper-parameters.

same machine learning model without text. We do make this comparison in Appendix F.2 and F.3, but it is not an entirely fair one. The machine learning algorithms are most suited to a large number of features and our experience of using them with a modest number of likely informative regressors is that they may do no better than OLS and so would not provide as difficult a benchmark to beat. Indeed, the results in Appendix F.2 and F.3 show that for every forecast test we do, the performance of the machine learning models with text is even stronger relative to a machine learning model (without text) benchmark, highlighting that OLS is harder to beat. Another reason for adopting OLS as our benchmark estimation method is that it is very widely used in practice.

An example forecast that uses machine learning, an artificial neural network, and term frequency vectors versus an OLS-AR(1) benchmark may be seen in Figure 7 for 3m-on-3m GDP growth at monthly frequency. In both cases, a single lag of GDP is included as a feature. Relative to the benchmark, the improvement in the goodness of fit is discernible in Figure 7, and the machine learning model also appears to be quicker to identify turning points.

In Figure 8 we show the forecast performance relative to the OLS-AR(1) benchmark for a range of machine learning models. As before, error bars are standard deviations over horizons of three to nine months ahead and across the different newspapers. In contrast to forecasts with the algorithmic text metrics (see Figure 5), there are performance improvements relative to the benchmark for every target variable.[13] The magnitude of these improvements is far larger too – in the previous section, few of the text metrics reached an improvement of 20 percentage points on any target variable while a substantial number of the machine learning forecasts with text have improvements of 30 percentage points or more. Two models perform consistently well: the shallow neural network and ridge regression. The performance of Lasso is very similar with and without text because it is not putting any weight on the term frequency vector of text information, and the elastic net performs similarly for similar reasons; Appendix H presents a variable importance exercise as evidence of this. Lasso is a sparse model, that selects a few variables to put weight on. In contrast, neural networks are not as constrained in how they use the features fed into them. This may suggest that dense models perform better, which would be consistent with Giannone, Lenza and Primiceri (2017), who find that models that put some weight on a broad set of macro variables do best at forecasting macro variables. Appendix F.1 presents further statistics on this model.

[13] It should be noted that not all of the ML models improve forecasting power. For example, Lasso and Elastic Net underperform consistently relative to the OLS benchmark. This suggests the success of ML is dependent on the task at hand.

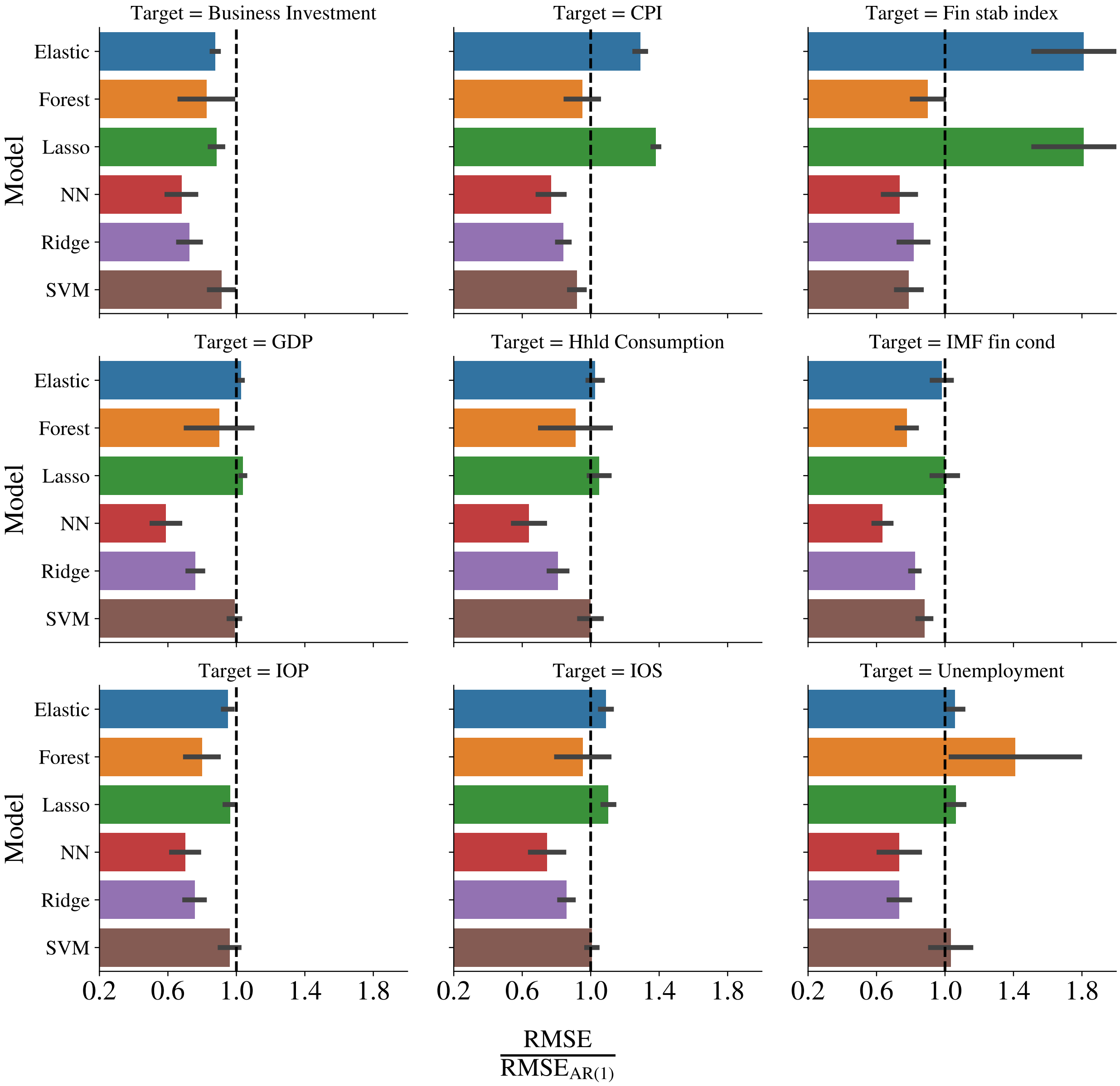

Figure 8: The relative improvement in root mean square error of a machine learning model that uses text and an AR(1) term versus OLS with the AR(1) term only. The facets are different target variables. Bars to the left of the dashed line indicate an improvement in forecast performance conducted with the given text metric. The confidence intervals are standard deviations over both the newspapers and the different horizons (3, 6 and 9 months ahead).

We now look at a more stringent test of text, as we did in the case of the algorithmic text metrics in §5.1. We examine whether text still adds value when the baseline model includes two macroeconomic factors derived from the same dataset comprising 33 macroeconomic series used in the previous subsection. The results of this are shown in Figure 9.

Unlike for the simpler text metrics, we find that the added value of text and machine learning persists even when comparing against the tougher OLS-AR(1)-factor model benchmark. Quantitatively, the improvements in forecast performance are smaller than versus an AR(1) alone, as would be expected given that there is other highly relevant information included in the benchmark factor model. However, significant forecast improvements are still achieved. With at least one machine learning model, forecasts for every target variable are improved versus the baseline. The support vector machine, shallow neural network, and ridge regression offer the best performance. These approaches comprehensively and consistently offer forecast improvements, even versus a rich factor model.

Paper	Model	Target Horizon	Business Investment	CPI	Fin stab index	GDP	Hhld Consumption	IMF fin cond	IOP	IOS	Unemployment
The Daily Mail	Forest	9	-2.26**				-2.27**				
	NN	3		-2.07**			-2.09**	-1.80*			
		6	-1.77*	-1.99**	-1.76*	-2.36**	-2.07**				-1.93*
		9	-1.85*	-2.06**		-2.24**	-2.19**			-2.17**	-2.11**
	Ridge	3	-2.51**				-2.21**	-1.85*	-2.04**		
		6	-2.20**	-1.98**	-1.77*		-2.06**				
		9	-2.79***				-2.64***			-1.71*	
	SVM	3	-1.96*		-1.79*		-1.88*	-2.10**			
		6			-1.90*						
		9		-2.16**							-2.03**
The Daily Mirror	Forest	6			-2.01**						
		9					-2.13**				
	NN	3		-1.84*				-1.68*			
		6		-1.87*	-1.66*	-2.70***	-2.04**			-2.08**	-2.19**
		9	-1.87*	-2.14**		-1.96*	-1.86*			-1.90*	-1.88*
	Ridge	3	-1.80*					-1.79*	-1.80*		
		6	-3.28***				-1.76*				
		9	-2.31**				-2.05**				
	SVM	9		-1.81*							
The Guardian	Elastic	3	-1.86*								
	Forest	9					-2.07**				
	Lasso	3	-1.81*								
	NN	3	-1.89*								
		6		-2.21**	-1.82*		-1.92*				-1.76*
		9		-2.12**			-2.21**				-2.66***
	Ridge	3	-2.17**	-1.89*				-1.91*			-2.28**
		6	-1.76*	-1.81*	-1.79*		-2.15**				
		9	-1.65*	-1.89*			-2.73***				
	SVM	3	-2.37**					-1.77*			
		9		-2.65***							-2.18**

Table 6: Results from a Diebold-Mariano test on the factor model using machine learning. Statistically significant differences in RMSE are shown. *, **, *** denote rejection of the null, of no difference in RMSE relative to an AR(1) and factors, at the 10%, 5%, 1% levels respectively. In the interest of space, only those targets for which at least one model-newspaper pair had a p-value of less than 10% are included.

One potential concern is that in running forecasts with so many methods, targets, horizons, and newspapers, our results may show forecast improvements that are statistical flukes. The error bars imply that this is not the case. To demonstrate this point more formally, we run a Diebold-Mariano test

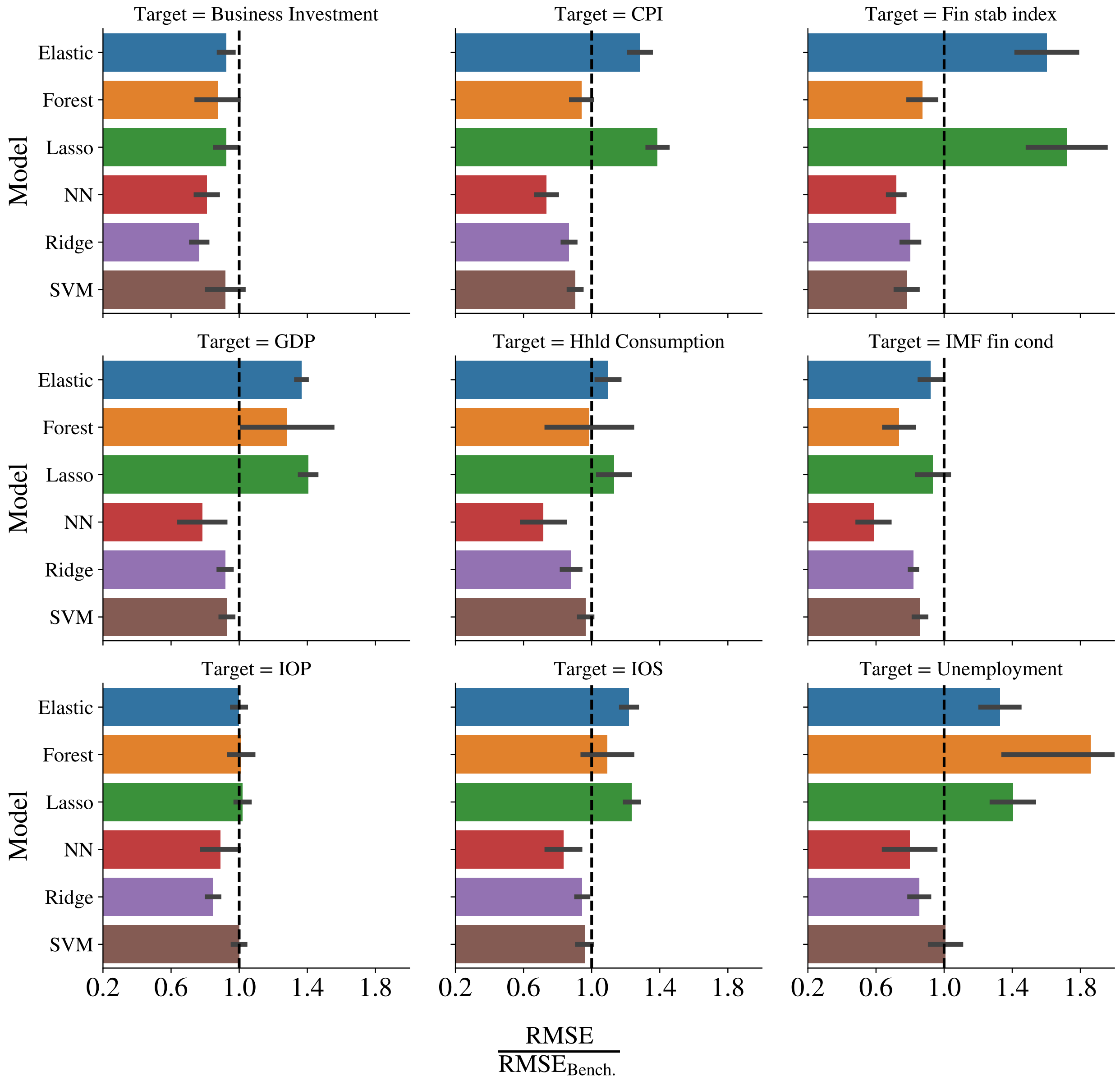

Figure 9: The relative improvement in root mean square error of a machine learning model that uses text, an AR(1) term, and factors versus OLS with the AR(1) and factors but no text. The facets are different target variables. Bars to the left of the dashed line indicate an improvement in forecast performance conducted with the given text metric. The confidence intervals are standard deviations over both the newspapers and the different horizons (3, 6 and 9 months ahead). There are good improvements in CPI, GDP, unemployment, investment, and consumption.

	target	Hhld Consumption	CPI	IMF fin cond	IOP	IOS	Unemployment	GDP	Fin stab index	Business Investment
horizon	type									
3	consistent	0.29	0.06	0.10	0.16	0.35	0.01	0.47	0.54	0.08
	lower	0.02	0.06	0.10	0.08	0.02	0.01	0.14	0.08	0.07
	upper	0.32	0.16	0.10	0.16	0.38	0.56	0.61	0.54	0.08
6	consistent	0.01	0.02	0.14	0.19	0.04	0.25	0.23	0.44	0.11
	lower	0.01	0.02	0.14	0.18	0.01	0.01	0.01	0.06	0.11
	upper	0.01	0.08	0.14	0.19	0.06	0.47	0.27	0.44	0.11
9	consistent	0.02	0.14	0.10	0.06	0.01	0.02	0.02	0.39	0.09
	lower	0.02	0.01	0.10	0.06	0.01	0.00	0.02	0.07	0.09
	upper	0.02	0.17	0.10	0.06	0.01	0.02	0.02	0.39	0.09

Table 7: Results from a test for superior predictive ability (Hansen, 2005; Sheppard et al., 2021) for machine learning versus an OLS-AR(1)-factor model. This test aims to determine whether, collectively, the alternative models outperform the benchmark model. The *lower* and *upper* p-values provide a lower and upper bound for the true *p-values* respectively. Rejection of the null indicates that the competing models are significantly better that the benchmark.

(Diebold and Mariano, 1995), with a small sample adjustment from Harvey, Leybourne and Newbold (1997), to check whether our results are statistically distinguishable from forecasts with the factor benchmark model in Table 6.[14]

Additionally, we apply the test for superior predictive ability (Hansen, 2005) which indicates whether the text models collectively outperform the benchmark models for the different targets and horizons. This test is useful here because we are comparing multiple alternative models against a benchmark. The results of this test for machine learning versus an OLS-AR(1)-factor model can be seen in Table 7, while the results of the test for machine learning versus an OLS-AR(1) model are in the Appendix, in Table F.8. We see rejections of the null (that the benchmark is superior) at the 5% level and for at least one forecast horizon for every series though some patterns are evident; the forecasts for the IMF Financial Conditions and Financial Stability indices are notably weak, and forecasts of GDP and its components are stronger at longer forecast horizons. This is consistent with the rest of our findings. Also consistent with the rest of our findings is that forecasts at longer horizons reject the null somewhat more often.

It is notable that we find relatively poor performance of random forests relative to the other machine learning models. This may in part be due to the forecasting scenario that we use here; random forests are by construction limited to predictions that are below (above) the historical maximum (minimum) of the target variables, but the period we look at includes a time of unprecedented change, namely the Great Financial Crisis, and the start of this period is also when most of the performance gains of the machine learning models occur relative to their respective benchmark models. Our results provide some

[14] Note that this test is still applicable to nested models in our case because we use rolling window estimation (Giacomini and White, 2006). We use the Newey-West estimator with the maximum order increasing with the horizon.

contrast with Borup and Schütte (2020), who found that random forests performed well as compared with an auto-regressive model in forecasting US employment growth, and with Medeiros et al. (2021), who found that random forests were the best of several models in forecasting US inflation (with ridge regression close behind). However, there are many differences between these studies: our data are different, the rolling window size is different, neither other study included neural nets as a model, and we use many more features (in the thousands as opposed to few hundreds). There is evidence that random forests perform less well when the set of predictors is very large and those predictors are weak as opposed to strong (Borup et al., 2020a). To be sure that the poor relative performance of random forests in our forecasting exercise is not due to hyper-parameter settings, we also perform a hyper-parameter tuning exercise for GDP, in our usual forecast configuration. The results (not shown) are consistent with the rest of this section; ridge and the shallow neural network are the strongest performers, while random forest does not beat the benchmark.

To provide further insight as to the performance of the Ridge, SVM, and Neural Network models, we run a variable importance exercise, which may be found in Appendix H. This confirms that all models predominantly put weight on the lag and the macro factors, but that these three best performing models tend to put weight on many (somewhat consistent) terms. In contrast, the Random Forest puts weight on a different set of words to these three, while the remaining models hardly put weight on text terms at all (and effectively put *no* weight on them for GDP, CPI, and unemployment). Based on this exercise, the ability to put a very small amount of weight on many text terms differentiates the more performant models. Note that the terms that are selected as the most important are plausibly related to economic phenomena: they include 'price', 'credit', 'financial', 'growth', 'government', 'inflation', 'labour', 'tax', 'house prices', and 'income'.

Many of our statistically significant forecast improvements occur at the 9 month horizon and, from the Diebold-Mariano tests, we see somewhat consistent gains in forecast improvements for investment and consumption. Both of these facts lend credence to the Shiller hypothesis that news influences views rather than the idea that news is simply a better real time source of information.

5.3 When does text improve forecast performance?

Having shown that various different methods allow text to contribute to forecasts, we ask: when does text count most for forecasts? While we do not present exhaustive evidence to answer this question, we do find evidence that the gains occur at specific points in time.

The breakdown of differences in squared error between OLS with only an AR(1) term and the most effective machine learning models with text and an AR(1) term are shown in Figure 10 on the left-hand axis and are denoted by $\varepsilon^2_{\text{Bench.}} - \varepsilon^2_{\text{Text}}$. Also shown is the forecasted variable, GDP growth, on the right-hand axis. For the three lines representing the squared error difference in forecast relative to the benchmark model, being above zero shows that a model with text is performing better than the benchmark model (which doesn't use text).

Figure 10 shows that most of the improvement in performance relative to the benchmark comes during the financial crisis and the period immediately following it. More generally, forecast improvements happen around turning points. We observe similar findings for other target variables, too.[15] The same pattern is seen with the best performing text metrics (Appendix G): text seems to tell us when macroeconomic trends are changing.

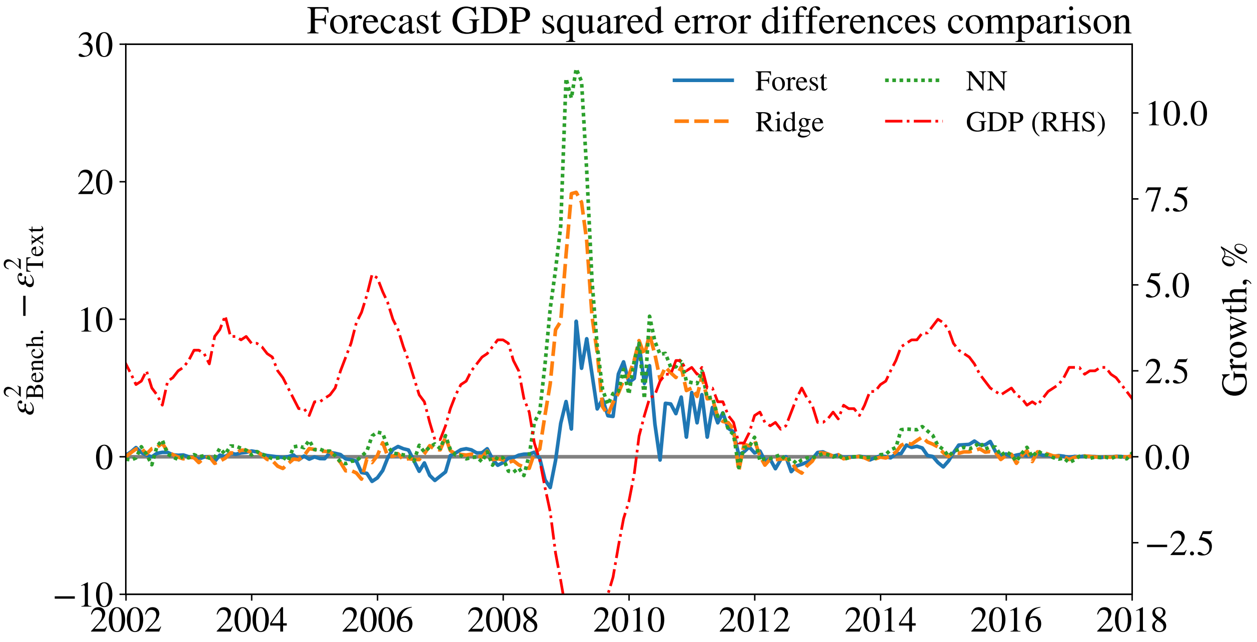

Figure 10: Mean squared error differences between a benchmark model and text. Errors are the average of the h-month ahead out-of-sample forecasts, with horizon $h = 3, 6, 9$. The target variable is monthly GDP, shown on the right-hand axis. The benchmark is an OLS AR(1) model. The plotted error bars are standard deviations over the different horizons and newspapers. For the squared error differences, a solid line above zero means that the model with text produces smaller errors than the benchmark model. Three different machine learning models are shown. The majority of the forecast gains are during the crisis.

[15] See, Figure F.4 for inflation and Figure G.5 for unemployment in the Appendix. In those cases, it is also evident the less strong performance of Forest across time.

6 Discussion, summary, and conclusion

We set out to discover whether newspaper text could provide information about future economic activity that is relevant to policymakers and, if it can, what methods might make text count the most towards that end. Our results show that, across a range of methods, text can indeed provide forward-looking information about key economic variables, and that this is robust both to horizons from 3 to 9 months and across three popular UK newspapers.

A key finding is that algorithm-based text indices that track sentiment or uncertainty can provide a similar signal to the macroeconomic and financial data that policymakers use today to track sentiment or uncertainty. Visual, correlational, and forecast-based evidence supports this. While much attention has previously focused on the extraction of information about uncertainty from text, we find that the signals of sentiment from text are much better correlated with proxies for sentiment.

However, we find an important distinction between pre-defined algorithms that turn text into time series and our novel machine learning and text approach; both are useful as forecast inputs on their own but only the latter produces better forecast performance once conditioning on the other macroeconomic data that would typically be available to policymakers. That is, while algorithm-based text metrics are useful on their own, they add little value when conditioning on the other macroeconomic and financial information that might be available to a policymaker when making a forecast. They are to some extent substitutes for the other information. In contrast, the combination of machine learning and text can provide additional, complementary value in forecasts–even when conditioning on the other non-text information that is available.

With regards to specific metrics and methods and their contributions, even a log transform of the counts of the term economy ('econom' to be precise) was able to add value in a simple forecasting model that does not condition on other information. This measure also correlates well with other proxies for economic activity. This should not be surprise – when newspapers talk about the economy a lot, it is likely to be because change is afoot. Common words with a clear meaning can carry strong signals. A simple count of the word uncertainty did almost as well as the more complex Boolean methods for uncertainty suggested by Alexopoulos and Cohen (2009) and Baker, Bloom and Davis (2016).

Second, we find that the dictionary method of Correa et al. (2017), originally designed to be an index for financial stability, performs the best of the wide range of algorithmic methods we tested both as a proxy for sentiment and as an input into forecasts when not conditioning on other information. As the newspapers used in our analysis are not geared towards specialists in financial markets, but towards the general public, this is a good indication of its general power to capture economic sentiment.

Finally, to get the most out of text, we find that the new approach we introduce – which retains thousands of terms from text and turns each into a time series that can be used with a machine learning model – is the most effective. It is a departure from simpler models that collapse article text into a single number. While this approach is not appropriate for the construction of an indicator, because the machine learning model learns as it goes, it produces the best improvements in forecasts in our tests: even when conditioning on other information. The reasons for this are likely that more of the text gets into the model, the model decides which terms to put more weight on, and the models we use to do this are known to be very powerful at prediction problems. The two machine learning models that perform the best are those that are able to put weight on many terms simultaneously, neural networks and ridge regression. While neural networks are able to capture non-linearity, ridge regression cannot, which suggests that the success of these two models may be due to them concurrently putting some weight on many features rather than their ability to capture non-linear relationship between variables.

Our method of using machine learning on term frequencies also has the advantage of being transferable to the prediction of any continuous variable using whatever text the researcher is interested in: that is, it is transferable to many other domains and applications away from the macroeconomic examples we present here. Showing that creating a large number of terms from text, turning each into a time series, and then applying sophisticated machine learning methods produces superior predictions across a wide range of target variables is one of our key contributions.

Policymakers make judgements about what statistical forecasts to put weight on, as well as assessing measures of sentiment, uncertainty, and a wide range of other quantitative and qualitative information, when formulating a central forecast. We have shown that text, and forecasts made with text, can be a useful addition to the range of information that policymakers use when making these judgements. Specifically, our results have an immediate application in policy situations where decisions must be made taking account of the near-term economic outlook but there is no official data, or even survey data, available on current conditions. In this circumstance, text can provide a more timely read on economic activity. For the three key macroeconomic time series – GDP, unemployment, and CPI – we show that our approach gives forecast improvements versus a factor model benchmark that are as large as 30 percentage points of the benchmark RMSE and are also statistically significant.

Furthermore, we find that newspaper text adds the most to forecasts during times of stress, and this is of particular value to macroeconomic policymakers because judgements made during stressed times are likely to be more important. These findings echo those of Garcia (2013) for stocks. It may suggest that newspaper articles report on developments in the economy first, or that the feedback loops between newspaper reports and real economic activity become more important during stressed times. Indeed there is evidence that periods of stress correspond to times of greater sensitivity to news (Akerlof and Shiller, 2010) and that newspapers can significantly influence their readers' views (Kennedy and Prat, 2017). Shiller (2017) has suggested that viral narratives play a causal role in economic activity, with newspapers potentially acting as spreaders of such epidemiological narratives. The effectiveness of news in improving forecasts of business investment and consumption over long horizons provides suggestive evidence for Shiller's hypothesis.

There are a number of avenues for future work. Here, we focused on predicting the first moment of our target variables rather than the second, but both are useful. Our findings also suggest that these methods could be applied to forecast economic turning points. We only exploited the timeliness of newspaper text here, not its high-frequency of information release–but this could also be useful for policymakers and the evidence that we present suggests that text could add value not just to near-term forecasts but to nowcasts too.[16]

[16] There is some evidence that it does – see `https://bankunderground.co.uk/2019/02/28/whats-in-the-news-text-based-confidence-indices-and-growth-forecasts/` and Borup et al. (2020b) for an example using text responses from survey data.

References

Akerlof, George A, and Robert J Shiller. 2010. Animal spirits: How human psychology drives the economy, and why it matters for global capitalism. Princeton University Press. 33

Alexopoulos, Michelle, and Jon Cohen. 2009. "Uncertain times, uncertain measures." University of Toronto Department of Economics Working Paper, 352. 8, 9, 15, 31, 39

Alexopoulos, Michelle, and Jon Cohen. 2015. "The power of print: Uncertainty shocks, markets, and the economy." International Review of Economics & Finance, 40: 8–28. 2

Antweiler, W., and M. Z. Frank. 2004. "Is all that talk just noise? The information content of internet stock message boards." Journal of Finance, 59(3): 1259–1294. 3

Ardia, David, Keven Bluteau, and Kris Boudt. 2019. "Questioning the news about economic growth: Sparse forecasting using thousands of news-based sentiment values." International Journal of Forecasting. 3

Bai, Jushan, and Serena Ng. 2002. "Determining the number of factors in approximate factor models." Econometrica, 70(1): 191–221. 22

Baker, Scott R, Nicholas Bloom, and Steven J Davis. 2016. "Measuring economic policy uncertainty." The Quarterly Journal of Economics, 131(4): 1593–1636. 2, 8, 14, 15, 31, 39

Bird, Steven, and Edward Loper. 2004. "NLTK: the natural language toolkit." 31, Association for Computational Linguistics. 38

Blei, David M, and John D Lafferty. 2006. "Dynamic topic models." 113–120, ACM. 6

Borup, Daniel, and Erik Christian Montes Schütte. 2020. "In search of a job: Forecasting employment growth using google trends." Journal of Business & Economic Statistics, 1–15. 2, 29

Borup, Daniel, Bent Jesper Christensen, Nicolaj Mühlbach, and Mikkel Slot Nielsen. 2020a. "Targeting predictors in random forest regression." SSRN 3551557. 29

Borup, Daniel, David E Rapach, Erik Christian Montes Schütte, et al. 2021. "Now-and backcasting initial claims with high-dimensional daily internet search-volume data." Department of Economics and Business Economics, Aarhus University. 47

Borup, Daniel, Jorge W Hansen, Benjamin Liengaard, and Erik CM Schütte. 2020b. "Quantifying investor narratives and their role during COVID-19." SSRN 3752116. 2, 33

Breiman, Leo. 2001. "Random forests." Machine learning, 45(1): 5–32. 23

Bybee, Leland, Bryan T Kelly, Asaf Manela, and Dacheng Xiu. 2020. "The structure of economic news." National Bureau of Economic Research. 4

Carriero, Andrea, Ana Galvão, and George Kapetanios. 2018. "A comprehensive evaluation of macroeconomic forecasting methods." International Journal of Forecasting (Forthcoming). 18, 44

Chang, Chih-Chung. 2011. "LIBSVM: a library for support vector machines." ACM Transactions on Intelligent Systems and Technology, 2:3(27). 22

Chatterjee, Somnath, Ching-Wai (Jeremy) Chiu, Sinem Hacioglu-Hoke, and Thibaut Duprey. 2017. "A financial stress index for the United Kingdom." Bank of England Staff Working Paper 697. 18, 19

Chauvet, Marcelle, and Simon Potter. 2013. "Forecasting output." In Handbook of Economic Forecasting. Vol. 2, 141–194. Elsevier. 43

Correa, Ricardo, Keshav Garud, Juan M Londono, Nathan Mislang, et al. 2017. "Constructing a Dictionary for Financial Stability." Board of Governors of the Federal Reserve System (US). 7, 8, 10, 16, 31, 39

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." arXiv preprint arXiv:1810.04805. 7

Diebold, Francis X, and Robert S Mariano. 1995. "Comparing predictive accuracy." Journal of Business & economic statistics, 20(1): 134–144. 28

Drucker, Harris, Christopher JC Burges, Linda Kaufman, Alex J Smola, and Vladimir Vapnik. 1997. "Support vector regression machines." 155–161. 22

D'Amuri, Francesco, and Juri Marcucci. 2017. "The predictive power of Google searches in forecasting US unemployment." International Journal of Forecasting, 33(4): 801–816. 2

Eckley, Peter. 2015. "Measuring economic uncertainty using news-media textual data." 5

Ellingsen, Jon, Vegard H Larsen, and Leif Anders Thorsrud. 2021. "News media versus FRED-MD for macroeconomic forecasting." Journal of Applied Econometrics. 3

Faust, Jon, and Jonathan H Wright. 2013. "Forecasting inflation." In Handbook of economic forecasting. Vol. 2, 2–56. Elsevier. 44

Friedman, Jerome H. 2001. "Greedy function approximation: a gradient boosting machine." Annals of statistics, 1189–1232. 47

Garcia, Diego. 2013. "Sentiment during recessions." The Journal of Finance, 68(3): 1267–1300. 3, 33

Gentzkow, Matthew, Bryan Kelly, and Matt Taddy. 2017. "Text as data." National Bureau of Economic Research. 2

Giacomini, Raffaella, and Halbert White. 2006. "Tests of conditional predictive ability." Econometrica, 74(6): 1545–1578. 28

Giannone, Domenico, Michele Lenza, and Giorgio E Primiceri. 2017. "Economic Predictions with Big Data: The Illusion Of Sparsity." C.E.P.R. Discussion Papers CEPR Discussion Papers 12256. 24

Gilbert, CJ Hutto Eric. 2014. "Vader: A parsimonious rule-based model for sentiment analysis of social media text." 8, 40

Greenwell, Brandon M, Bradley C Boehmke, and Andrew J McCarthy. 2018. "A simple and effective model-based variable importance measure." arXiv preprint arXiv:1805.04755. 47

Hansen, Peter Reinhard. 2005. "A test for superior predictive ability." Journal of Business & Economic Statistics, 23(4): 365–380. 28, 52

Harvey, David, Stephen Leybourne, and Paul Newbold. 1997. "Testing the equality of prediction mean squared errors." International Journal of forecasting, 13(2): 281–291. 28

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media. 23

Hoerl, Arthur E, and Robert W Kennard. 1970. “Ridge regression: Biased estimation for nonorthogonal problems.” Technometrics, 12(1): 55–67. 22

Hu, Guoning, Preeti Bhargava, Saul Fuhrmann, Sarah Ellinger, and Nemanja Spasojevic. 2017. “Analyzing Users’ Sentiment Towards Popular Consumer Industries and Brands on Twitter.” 381–388, IEEE. 8, 10, 39, 40

Hu, Minqing, and Bing Liu. 2004. “Mining and summarizing customer reviews.” 168–177, ACM. 8, 10, 39, 40

Husted, Lucas F., John Rogers, and Bo Sun. 2017. “Monetary Policy Uncertainty.” Board of Governors of the Federal Reserve System (U.S.) International Finance Discussion Papers 1215. 8, 15, 39

Jegadeesh, Narasimhan, and Di Wu. 2013. “Word power: A new approach for content analysis.” Journal of Financial Economics, 110(3): 712–729. 2

Jurado, Kyle, Sydney C Ludvigson, and Serena Ng. 2015. “Measuring uncertainty.” The American Economic Review, 105(3): 1177–1216. 12, 17

Kelly, Bryan T, Asaf Manela, and Alan Moreira. 2019. “Text Selection.” National Bureau of Economic Research Working Paper 26517. 4

Kennedy, Patrick, and Andrea Prat. 2017. “Where Do People Get Their News?” Columbia Business School Research Papers 17-65. 33

Keynes, John Maynard. 1936. The general theory of employment, interest, and money. Springer. 1

Kozeniauskas, Nicholas, Anna Orlik, and Laura Veldkamp. 2018. “What are uncertainty shocks?” Journal of Monetary Economics, 100: 1 – 15. 15

Larsen, Vegard H, and Leif A Thorsrud. 2019. “The value of news for economic developments.” Journal of Econometrics, 210(1): 203–218. 3

Loughran, Tim, and Bill McDonald. 2011. “When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks.” The Journal of Finance, 66(1): 35–65. 2

Loughran, Tim, and Bill McDonald. 2013. “IPO first-day returns, offer price revisions, volatility, and form S-1 language.” Journal of Financial Economics, 109(2): 307–326. 2, 7, 8, 10, 39

Manela, Asaf, and Alan Moreira. 2017. “News implied volatility and disaster concerns.” Journal of Financial Economics, 123(1): 137–162. 3, 4

Medeiros, Marcelo C, Gabriel FR Vasconcelos, Álvaro Veiga, and Eduardo Zilberman. 2021. “Forecasting inflation in a data-rich environment: the benefits of machine learning methods.” Journal of Business & Economic Statistics, 39(1): 98–119. 29

Mumtaz, Haroon, and Alberto Musso. 2018. “The evolving impact of global, region-specific and country-specific uncertainty.” European Central Bank Working Paper Series 2147. 15

Newsworks. 2018. “Circulation of newspapers in the United Kingdom (UK) as of June 2018 (in 1,000 copies).” Statista, Retrieved August 30, 2018. https://www.statista.com/statistics/529060/uk-newspaper-market-by-circulation/. 5

Nielsen, Finn Årup. 2011. “A new ANEW: Evaluation of a word list for sentiment analysis in microblogs.” arXiv preprint arXiv:1103.2903. 7, 8, 10, 39

Nothman, Joel, Hanmin Qin, and Roman Yurchak. 2018. "Stop Word Lists in Free Open-source Software Packages." 7–12. 38

Nyman, Rickard, Sujit Kapadia, David Tuckett, David Gregory, Paul Ormerod, and Robert Smith. 2018. "News and narratives in financial systems: exploiting big data for systemic risk assessment." Bank of England Staff Working Papers, 704. 3, 7, 8, 10, 39

Puurula, Antti. 2013. "Cumulative progress in language models for information retrieval." 96–100. 38

Rambaccussing, Dooruj, and Andrzej Kwiatkowski. 2020. "Forecasting with news sentiment: Evidence with UK newspapers." International Journal of Forecasting. 3

Redl, Chris. 2017. "The impact of uncertainty shocks in the United Kingdom." Bank of England Bank of England Staff Working Papers. 22, 48

Redl, Chris. 2018. "Uncertainty matters: evidence from close elections." Bank of England Bank of England Staff Working Papers. 12, 17

Rumelhart, David E, Geoffrey E Hinton, and Ronald J Williams. 1985. "Learning internal representations by error propagation." California Univ San Diego La Jolla Inst for Cognitive Science. 23

Shapiro, Adam Hale, Moritz Sudhof, and Daniel Wilson. 2018. "Measuring news sentiment." Federal Reserve Bank of San Francisco. 3

Sheppard, Kevin, Stanislav Khrapov, Gábor Lipták, mikedeltalima, Rob Capellini, Hugle, esvhd, Alex Fortin, JPN, Weiliang Li, Austin Adams, jbrockmendel, M. Rabba, Michael E. Rose, Tom Rochette, UNO Leo, Xavier RENE-CORAIL, and syncoding. 2021. "bashtage/arch: Release 5.0.1." "v5.0.1." 28, 52

Shiller, Robert J. 2017. "Narrative economics." The American Economic Review, 107(4): 967–1004. 1, 11, 33

Tetlock, Paul C. 2007. "Giving content to investor sentiment: The role of media in the stock market." The Journal of finance, 62(3): 1139–1168. 3, 7, 8, 10, 39

Thorsrud, Leif Anders. 2018. "Words are the new numbers: A newsy coincident index of the business cycle." Journal of Business & Economic Statistics, 1–17. 3, 4, 6

Tibshirani, Robert. 1996. "Regression shrinkage and selection via the lasso." Journal of the Royal Statistical Society. Series B (Methodological), 267–288. 22

Zipf, George K. 1950. "Human behavior and the principle of least effort." 8

Zou, Hui, and Trevor Hastie. 2005. "Regularization and variable selection via the elastic net." Journal of the royal statistical society: series B (statistical methodology), 67(2): 301–320. 22

Making text count: economic forecasting using newspaper text

Appendix

Eleni Kalamara Arthur Turrell Chris Redl George Kapetanios Sujit Kapadia

A Text cleaning

Text must be processed in order for it to be used in any quantitative application. Except where stated otherwise, for the algorithm-based text metrics, we use the following methods to pre-process newspaper text.

1. remove punctuation, hyperlinks, hyper text markup language (HTML) tags, special characters, leading or trailing white space characters, and digits;

2. set all characters in lower case; and

3. drop words which are in our list of stop words.

Note that we do not use stemming or lemmatisation. It is common practice to drop a large number of words from a corpus before turning it into a quantitative measure over text. One of the reasons is that a large number of words in any text corpus is uninformative, either because it occurs very rarely or very frequently. Words in the latter category are often known as ‘stop words’ and include ‘and’, ‘is’, ‘in’, and so on (see Nothman, Qin and Yurchak (2018) for a discussion). As noted in §3, one of the common approaches to excluding words is to use threshold frequencies (both high and low) applied to the entire corpus. However, this requires knowledge of the entire corpus ahead of time and is not suitable for real time forecasting. Instead, it is necessary to define ahead of time a set of words that will not be retained. We drop words from the union of two popular lists of stop words: the NLTK word list (Bird and Loper, 2004) and the list proposed by Puurula (2013).

B Turning text into time series

B.1 Algorithm based text metrics

B.1.1 Dictionary methods

Dictionary methods measure sentiment using a pre-defined list of words associated with scores. The scores are usually positive and negative scores with values of +1 and -1, respectively, in the simplest case. These scores are counted for each article. The net score, weighted by the number of words with scores, is the sentiment score for each article. For each news source, let articles – which consist of a group of (possibly repeated) terms – be denoted a. Each dictionary D is split into positive, D^+, and negative, D^- parts and defines a mapping $D : W \to C$ such that $w \in W$ has an associated score $c \in C$. Not all terms in every article are in the domain of D. The sentiment score for an article a with terms w is given by

$$S = \frac{1}{|w|}\left(\sum_{w} D^+(w) - \sum_{w} D^-(w)\right)$$

Table B.1: Lists of words in the UK BBD metric.

E, Economics words	economic, economy
U, Uncertainty words	uncertainty, uncertain
P, Policy words	spending, policy, deficit, budget, tax, regulation, bank of england

The purely dictionary based text metrics with positive and negative words which we use are from Nyman et al. (2018), Loughran and McDonald (2013), Nielsen (2011), Hu and Liu (2004) and Hu et al. (2017), and Correa et al. (2017) in addition to the Harvard IV psychological dictionary used by Tetlock (2007).

B.1.2 Boolean methods

These metrics are typically counts of articles which satisfy a logical condition (within a given time period). They may also be normalised by the total number of articles within a time period. As the most simple example of this, we count the number of occurrences of "uncertain" and "econom" aggregated over the relevant time scale.

We also use more elaborate Boolean metrics. For instance, ones in which, given two sets of words, E and U, and w a term in article a, article a is counted if and only if

$$(w \in E) \wedge (w' \in U) \quad \forall \quad w, w' \in a$$

A daily measure is created from the ratio of the number of counts each day to the number of articles satisfying the condition each day.

The uncertainty measure of Alexopoulos and Cohen (2009) falls into this category, with $U = \{\text{uncert}, \text{uncertainty}\}$ and $E = \{\text{econom}, \text{economy}\}$.

Baker, Bloom and Davis (2016) describe 'Economic Policy Uncertainty'. The UK measure uses counts of the logical combination of three lists. We use a very similar measure to theirs, denoted as 'baker_bloom_davis'. If terms from all three of the lists shown in Table B.1 appear in an article, a count is recorded.

We also use the Boolean logic monetary policy uncertainty measure of Husted, Rogers and Sun (2017) with a slight modification for real time forecasting. Their measure counts the number of articles containing the triple of (i) "uncertainty" or "uncertain," and (ii) "monetary policy(ies)" or "interest rate(s)" or "Bank rate" and (iii) " Bank of England" or "BoE". This is normalised by the total number of articles mentioning category (iii) words for a given newspaper-period. The index is then rescaled to have a standard deviation of unity across the entire sample. For our purposes, the latter step is not appropriate as it introduces information leakage. Instead, where we normalise, we only use data up to and including that point, or the in-sample in a forecast test environment. We divide by the number of articles mentioning category (iii) words within each day.

B.1.3 Word counts

We include in our text metrics some simple counts of the number of words, and also transforms of those simple counts. We use two metrics that are transforms of counts: TFIDF economy and TFIDF uncert which, as part of their construction, look for the strings 'econom' and 'uncertain' respectively. The details of the tfidf transforms are in §3.1.

B.1.4 Methods from computer science

We use the Valence Aware Dictionary for sEntiment Reasoning (VADER) metric of Gilbert (2014). This is a rule based metric that embodies grammatical and syntactical conventions that humans use when expressing or emphasising sentiment intensity. It is oriented to small snippets of text, such as tweets, and produces a magnitude of sentiment in addition to a sign. The (unnormalised) sentiment intensity is on a scale from -4 to +4. For example, the word "okay" has a positive score of 0.9, "good" is 1.9, and "great" is 3.1, whereas "horrible" is -2.5, and the frowning emoticon ":(" is -2.2. The sentiment scores are calculated on a sentence level and we create per article sentiment by averaging the scores and dividing by the total number of sentences in each article[17].

We also adopt a metric based on the opinion mining literature (Hu et al., 2017; Hu and Liu, 2004). Although strictly speaking a dictionary method, the words have not been selected *a priori* by a researcher. Instead, the 'opinion sentiment' dictionary is constructed from words which have strong positive or negative connotations as discovered by text summarisation techniques applied to web reviews of products. As such, the dictionary reflects consumer preferences. The series are constructed by subtracting the positive and negative counts of words and normalising by the total number of words in each article.

B.2 Stationarity of text metrics

We determine whether our algorithm-based series are stationary. An augmented Dickey-Fuller test was run, using the Akaike information criterion to choose the number of lags, to test the null hypothesis of a unit root against the alternative hypothesis of stationarity. At a 1% significance level, we can reject the null hypothesis for all metrics for at least one of the three newspapers. The null cannot be rejected at the 10% significance level for a small number of newspaper-text metric pairs, mostly those based on raw counts of occurrences. *The Guardian* had the fewest significant results. The null is rejected for a wide range of metrics.

B.3 Granger Causality Tests

Tables B.3 and B.4 show results of Granger causation tests with text metrics and proxies for both sentiment and uncertainty.

C Forecast environment

Features are indexed by $k = 0, \ldots, K$, time by $t = 0, \ldots, T$, the window step size by $s \geq 1$, the initial training period length as $\alpha + s$, and train (and associated test) periods by $\mu = 1, \ldots, \frac{T-s-\alpha}{s}$. $\alpha = 0$ implies that the initial training period is of length s. Define $\{y_t\}_{t=0}^{t=T}$ as the target variable shifted h

[17] The model is available as a part of NLTK sentiment analysis Python package.

	The Daily Mirror	No. obs.	The Daily Mail	No. obs.	The Guardian	No. obs.
TFIDF uncert	-03.37**	254	-07.66***	272	-04.22***	318
Counts uncert	-01.28	242	-04.05***	268	-01.79	308
Alexopoulos	-01.87	241	-04.14***	267	-01.70	309
Baker-Bloom-Davis	-05.81***	255	-05.29***	271	-00.99	309
Husted	-08.70***	256	-08.97***	272	-04.03***	319
Opinion	-04.35***	254	-04.56***	269	-03.09**	320
Harvard	-07.67***	255	-03.08**	264	-04.01***	319
Loughran	-04.61***	255	-04.43***	268	-02.16	320
Vader	-02.78*	251	-02.95**	267	-03.13**	320
Afinn	-02.63*	251	-03.27**	268	-02.99**	320
Counts economy	-01.96	249	-03.46***	270	-03.31**	320
Stability	-04.47***	255	-04.90***	270	-04.47***	321
TFIDF economy	-02.87*	255	-02.69*	264	-03.36**	311
Nyman	-05.60***	255	-04.54***	271	-03.45***	311

Table B.2: Results of an Augmented Dickey-Fuller test on all text metrics. The number of observations differ as the number of lags to include is chosen using the AIC information criterion. Asterisks denote p-values; 1%: ***, 5%: **, 10%: *.

	Husted	Stability	TFIDF economy	Counts economy	Alexopoulos	Baker-Bloom-Davis	Counts uncert	Harvard	TFIDF uncert	Vader	Afinn	Nyman	Loughran	Opinion
BoE agg credit spread	45.95***	4.58***	2.57*	1.91	1.42	2.72**	2.69**	0.13	2.53*	2.65**	2.35*	1.34	1.08	0.72
Lloyds Bus Activity	0.82	7.20***	12.81***	12.07***	2.25*	2.70**	1.60	2.02	1.38	0.98	0.49	1.60	0.99	0.93
OECD Bus Conf	31.62***	2.08	1.94	1.69	1.61	0.71	0.35	0.66	0.15	0.97	1.03	1.15	2.46*	0.51
VFTSEIX	1.42	3.27**	0.83	1.04	3.76**	0.88	2.75**	4.94***	1.41	4.69***	4.75***	2.73**	2.11*	3.73**
Jurardo Macro uncert	5.46***	3.10**	0.67	1.10	4.01***	1.19	3.53**	1.06	4.70***	1.06	0.78	1.16	0.93	0.92
Lloyds Bus Conf	0.76	3.28**	7.03***	5.47***	0.73	0.88	0.50	1.92	0.57	1.38	1.92	1.39	1.46	1.55
IG Corp Bond spread	0.69	4.26***	4.62***	4.03***	2.28*	1.85	2.78**	1.17	1.85	0.38	0.90	0.55	1.09	1.89
Composite PMI	0.76	5.75***	2.10	2.52*	1.55	1.34	1.12	3.33**	1.59	1.45	1.14	1.75	0.12	0.43
Jurardo Fin uncert	3.37**	1.46	0.93	1.75	1.19	3.49**	0.74	1.12	0.87	0.78	0.41	0.07	0.73	0.13
GDP forecast std dev	2.14*	2.14*	0.64	0.33	2.56*	1.69	1.32	0.71	0.76	0.43	0.32	1.31	0.55	0.59
GfK Consumer Conf	2.69**	1.52	0.72	1.32	1.84	1.17	2.08	0.42	0.78	0.39	0.52	0.44	0.33	0.45
ERI volatility	0.44	1.42	0.86	1.35	1.20	2.98**	0.54	0.81	0.85	1.81	0.60	0.15	0.84	0.25
BoE uncert	1.96	2.64**	1.15	1.37	0.69	0.81	0.83	0.72	0.98	0.10	0.13	0.43	0.52	0.21
VIX	0.60	0.63	0.14	0.28	0.76	1.47	0.83	0.36	0.35	0.09	0.21	0.19	0.25	0.30

Table B.3: Test of whether text metrics Granger cause proxies, at a three month horizon. The text metrics are averaged across the three newspapers. Asterisks denote p-values; 1%: ***, 5%: **, 10%: *.

steps ahead, for h the desired horizon of the forecast. It is denoted $\vec{y}$ for short. Let $\{x_{tk}\}_{t=0}^{t=T}$ represent feature k, also denoted $\vec{x}_k$. The entire set of features of all time form a matrix X. Though we use rolling window estimation for all results presented we define below the cuts of the data for both expanding and rolling window estimation. Also, in both cases, the test set is composed of data points that have never been used for estimation and lie in the future (in time) of the training set, i.e. for a rolling window from $t = 20$ to $t = 25$ the test set would run from $t = 26$ to $t = T$.

Our in-sample and out-of-sample results as presented are created from the union of the last in-sample prediction of each estimation window and the first out-of-sample prediction of the same estimation window, respectively. These are defined formally below.

C.1 Expanding window

Define

$$I^e_\mu(\vec{z}) = \left\{ z_t \right\}_{t=0}^{t=\mu \cdot s+\alpha-1}$$

	VFTSEIX	Lloyds Bus Conf	IG Corp Bond spread	BoE uncert	BoE agg credit spread	Jurardo Macro uncert	Composite PMI	GfK Consumer Conf	OECD Bus Conf	Lloyds Bus Activity	GDP forecast std dev	Jurardo Fin uncert	ERI volatility	VIX
TFIDF economy	4.30***	5.10***	8.52***	4.05***	5.17***	6.74***	3.06**	4.57***	5.19***	0.98	3.19**	2.17*	0.31	0.94
Stability	11.99***	6.38***	5.57***	3.72**	3.99***	3.02**	2.24*	2.02	3.45**	2.13*	1.63	1.05	0.73	2.95**
Counts economy	3.83**	5.26***	6.04***	2.84**	4.87***	4.13***	2.86**	2.88**	3.32**	1.49	2.06	1.89	0.28	0.26
Husted	2.69**	5.18***	0.33	2.87**	5.89***	1.86	0.50	0.37	0.33	5.18***	0.23	6.34***	4.20***	0.46
Alexopoulos	8.85***	4.46***	3.53**	4.55***	1.53	0.74	1.14	1.36	0.64	0.93	1.07	0.37	1.98	0.53
Vader	6.02***	3.87**	2.40*	2.45*	0.97	3.01**	4.01***	1.68	0.66	0.98	1.55	1.20	0.92	1.13
Baker-Bloom-Davis	8.12***	2.98**	1.26	3.18**	3.27**	1.38	0.49	2.65**	3.35**	0.87	0.50	0.82	1.61	0.34
Loughran	1.69	2.06	3.58**	2.43*	1.52	2.82**	3.95***	2.72**	1.06	1.78	1.27	1.24	0.78	0.59
Harvard	2.30*	3.65**	4.02***	1.28	2.30*	1.36	3.09**	0.76	0.16	1.40	1.53	0.21	1.07	1.32
Afinn	2.17*	1.97	3.28**	1.99	1.03	2.37*	4.04***	0.98	0.88	1.12	1.57	0.67	1.05	1.21
Opinion	4.12***	0.99	2.23*	1.01	0.92	1.97	2.38*	0.64	0.91	0.60	1.98	0.64	0.76	0.76
Counts uncert	4.99***	2.83**	0.76	3.06**	0.76	0.11	0.34	1.44	0.55	0.05	0.52	0.92	1.57	0.27
Nyman	3.44**	2.76**	1.87	0.34	0.76	0.13	1.40	0.03	0.24	1.04	0.77	0.60	0.92	0.67
TFIDF uncert	4.84***	2.17*	0.87	1.45	2.15*	0.14	0.09	0.54	0.37	0.15	0.76	0.04	0.82	0.50

Table B.4: Test of whether proxies Granger cause text, at a three month horizon. The text metrics are averaged across the three newspapers. Asterisks denote p-values; 1%: ***, 5%: **, 10%: *.

as the in-sample expanding window slice μ for an arbitrary time vector $\vec{z}$. Similarly, define the associated out of sample slice μ as:

$$O^e_\mu(\vec{z}) = \Big\{ z_t \Big\}_{t=\mu\cdot s+\alpha}^{t=T}$$

Transformations T are labelled by whether they are expanding (e) or rolling (r), and for the feature, k, they are based on. For instance, a normalisation transformation is given by

$$T^e_{\mu k}(\vec{z}) = T^e_{\mu k}(\vec{z}; I^e_\mu(\vec{x}_k)) = T^e_{\mu k}\left(\vec{z}; \{x_{kt}\}_{t=0}^{t=\mu\cdot s+\alpha-1}\right) = \frac{\vec{z} - \langle I^e_\mu(\vec{x}_k)\rangle}{\sigma_{I^e_\mu(\vec{x}_k)}}$$

Transformations are indexed by μ to avoid information leakage (aka look-ahead bias). In general, the feature index on T will be implicit.

Define f_μ as the model which results from trying to fit $T^e_\mu(I^e_\mu(X))$ to $\vec{y}$. In-sample tests are based on $f_\mu\left(T^e_\mu(I^e_\mu(X))\right)$, while out-of-sample tests are performed on

$$f_\mu\left(T^e_\mu(O^e_\mu(X))\right)$$

To create a unified in-sample set from the end of each in-sample estimation window (recall that each these is indexed by μ), take

$$\mathcal{I}^e = \bigcup_\mu \Big\{ f_\mu\left(T^e_\mu(I^e_\mu(X))\right) \Big\}_{t=(\mu-1)s+\alpha}^{t=\mu s-1+\alpha}$$

This takes, for each possible value of t, the model prediction with the index label that has the highest possible value of μ. The final test, or out-of-sample, set that we use is constructed similarly: for each possible value of t, it is the model prediction with the lowest possible value of μ:

$$\mathcal{O}^e = \bigcup_\mu \Big\{ f_\mu\left(T^e_\mu(O^e_\mu(X))\right) \Big\}_{t=\mu s+\alpha}^{t=(\mu+1)s-1+\alpha}$$

Equivalently, the in-sample and out-of-sample sets are composed of the last step of each training window indexed by μ, and the first step of each test set indexed by μ.

C.2 Rolling Window

A window of size $\alpha + s$ is used to estimate the model.

$$I^r_\mu(\vec{z}) = \left\{ z_t \right\}_{t=(\mu-1)\cdot s}^{t=\mu\cdot s+\alpha-1}$$

$$O^r_\mu(\vec{z}) = \left\{ z_t \right\}_{t=\mu\cdot s+\alpha}^{t=T}$$

$$T^r_{\mu k}(\vec{z}) = T^r_{\mu k}(\vec{z}; I^r_\mu(\vec{x}_k)) = \frac{\vec{z} - \langle I^r_\mu(\vec{x}_k)\rangle}{\sigma_{I^r_\mu(\vec{x}_k)}}$$

The unified, one-step ahead dataset is created from

$$\mathcal{I}^r = \bigcup_\mu \left\{ f_\mu \left(T^r_\mu(I^r_\mu(X)) \right) \right\}_{t=(\mu-1)s+\alpha}^{t=\mu s-1+\alpha}$$

and

$$\mathcal{O}^r = \bigcup_\mu \left\{ f_\mu \left(T^r_\mu(O^r_\mu(X)) \right) \right\}_{t=\mu s+\alpha}^{t=(\mu+1)s-1+\alpha}$$

Note that, because the global transformations depend on the training data, $\mathcal{I}^r \neq \mathcal{I}^e$ and $\mathcal{O}^r \neq \mathcal{O}^e$.

In the forecasting exercises, we use $\alpha = 36$ and $s = 1$ where $\alpha + s = 37$ is the full window size and s is the step-size that the window moves for each forecast.

D Algorithm-based text metrics – further forecast results

This section present further results related to §5.1.

D.1 Performance versus an AR(1) model benchmark

For the case in which the benchmark model for the algorithmic text based metrics is an AR(1), we run a Diebold-Mariano test to check whether the results are statistically distinguishable from forecasts with the benchmark model. In the table, we show only those forecasts for which at least one target-metric combination per newspaper had a statistically significantly smaller RMSE than the benchmark model and we look at $h = 9$. We find statistically significant results across newspapers, although *The Guardian* does more poorly than the other two. The most consistent pattern of forecast performance is across targets. Although the gains for CPI look small in Figure 5, Table D.5 makes it clear that they are significant for some combinations of newspaper, metric, and target.

D.2 Performance versus a factor model benchmark

In Table D.6 we present results from a Diebold-Mariano test for a model including a text metric, two factors, and an AR(1) versus the same model without the text metric at $h = 9$. Combinations of targets and metrics that were not statistically significant with the simpler AR(1) model do reach statistical significance in this test, somewhat counter-intuitively. However, this is not an unusual finding. Several studies drawn from the forecasting literature suggest that univariate time series models have better forecasting power than richer models, especially for macroeconomic time series (Chauvet and Potter,

Paper	Metric	Target Horizon	Business Investment	CPI	Fin stab index	GDP	Hhld Consumption	IMF fin cond	IOP	Unemployment
The Daily Mail	Afinn	3								-1.72*
		6	-1.69*							-1.65*
	Alexopoulos	9	-1.67*							
	Counts economy	9					-2.04**			
	Husted	9							-2.25**	
	Loughran	6				-1.78*				
		9	-1.91*			-1.72*	-1.93*			
	Nyman	3			-2.14**			-1.79*		
		9				-1.77*	-1.68*			
	Opinion	3			-1.93*					-1.67*
		6	-1.87*			-1.71*				
		9	-1.69*			-1.66*				
	Stability	3			-1.99**					
		6				-1.67*				
		9	-1.65*			-1.68*				
	TFIDF economy	3	-2.44**							
		9				-1.68*	-2.47**			
	TFIDF uncert	9			-1.69*					
	Vader	3								-1.74*
		6								-1.67*
The Daily Mirror	Afinn	6		-1.99**						
		9		-1.96*						
	Baker-Bloom-Davis	9		-1.76*						
	Harvard	6		-1.73*						
		9		-2.15**						
	Loughran	6		-1.83*						
	Nyman	3		-1.97**						
		6		-2.10**						
		9		-1.74*						
	Opinion	6		-1.66*						
		9		-2.64***						
	TFIDF economy	3	-1.72*			-1.84*				
	TFIDF uncert	6	-3.30***							
	Vader	6		-2.01**						
		9		-2.15**						
The Guardian	Afinn	3								-1.89*
	Alexopoulos	3								-1.90*
	Counts economy	3								-1.75*
		6								-1.73*
	Harvard	3								-1.79*
	Loughran	3								-1.88*
	Stability	3								-2.09**
	TFIDF economy	3								-1.96*
		6								-1.93*

Table D.5: Results from a Diebold-Mariano test of an OLS-AR(1) model with text metrics versus an AR(1) model without them (the benchmark). Statistically significant differences in RMSE are shown. *, **, *** denote rejection of the null, of no difference in RMSE relative to the benchmark model, at the 10%, 5%, 1% levels respectively. Only those targets for which at least one metric-newspaper pair had a p-value of less than 10% are included.

2013; Faust and Wright, 2013). Including more information does not necessarily improve forecasts at a horizon longer than one quarter. In particular, Carriero, Galvão and Kapetanios (2018) show that the choice of the best forecasting model class may vary with the forecast horizon.

Which targets can be forecast better using text are somewhat consistent with the AR(1) model: business investment, CPI, and household consumption feature heavily.

In the case of the factor model, we also see significant results for unemployment, and less strong results for GDP. As GDP is a composite measure, and the factors are designed to track many variables that go into its construction, this is unsurprising.

E Machine learning models

Here we present the specifications of the machine learning models and their hyper-parameters. For the implementation of the models we use the *scikit-learn* package (version 0.23.1), available in the Python programming language. We use the default *scikit-learn* settings for most hyper-parameters, as these

Paper	Metric	Target Horizon	Business Investment	CPI	Fin stab index	GDP	Hhld Consumption	IMF fin cond	Unemployment
The Daily Mail	Afinn	9					-2.10**		-1.81*
	Alexopoulos	9					-2.98***		
	Baker-Bloom-Davis	9					-2.94***		-1.91*
	Counts uncert	9		-2.27**					
	Harvard	9	-1.68*						
	Husted	9		-2.18**			-3.23***		
	Loughran	9							-1.79*
	Nyman	3			-2.00**			-1.95*	
		9					-2.36**		
	Opinion	3			-2.13**				
		6	-1.97**		-1.77*		-1.67*		-1.73*
		9					-2.10**		-1.88*
	Stability	3			-1.73*				
		6							-1.68*
		9					-2.05**		-2.09**
The Daily Mirror	Afinn	9		-1.69*					-2.05**
	Alexopoulos	3	-1.67*						
	Baker-Bloom-Davis	9							-1.93*
	Counts economy	9	-2.29**						
	Harvard	9		-1.77*					
	Opinion	9							-1.69*
	Vader	9		-1.71*		-2.05**	-1.84*		-1.78*
The Guardian	Afinn	3	-1.70*						
	Alexopoulos	3							-1.73*
		6							-2.54**
		9		-1.96*					-2.18**
	Baker-Bloom-Davis	6							-2.08**
		9							-2.17**
	Counts uncert	9		-1.84*					
	Harvard	3	-1.75*						
		6	-1.66*						
	Loughran	3	-1.74*						
	Nyman	3	-1.74*						
	Opinion	3	-1.74*						
	Stability	3	-1.91*						

Table D.6: Results from a Diebold-Mariano test on the factor model with algorithm-based text metrics. Statistically significant differences in RMSE are shown. *, **, *** denote rejection of the null, of no difference in RMSE relative to an AR(1) and factors (the benchmark model), at the 10%, 5%, 1% levels respectively. Only those targets for which at least one metric-newspaper pair had a p-value of less than 10% are included.

are based on expert experience and are designed to behave well in a wide variety of situations. We state the values of hyper-parameters used in this section.

Throughout, let $\{x_{tk}\}_{t=0}^{t=T}$ represent feature k, also denoted $\vec{x}_k$, and the entire set of features of all time form a matrix X. The features in X are typically the counts of text terms plus one lag of the target variable; sometimes there are also the factors from a factor model. More information on the construction of X may be found in §3. The time series of the target variable is denoted $\vec{y}$. Define

$$\|\beta\|_p = \left(\sum_{k=1}^{K} |\beta_k|^p \right)^{1/p}$$

as the ℓ^p norm.

E.1 Lasso

The least absolute shrinkage and selection operator solves

$$\min_{\beta} \left\{ \frac{1}{T} \|y - X\beta\|_2^2 \right\} \text{ subject to } \|\beta\|_1 \leq \kappa$$

with $\kappa = 1$.

E.2 Ridge

Ridge regression solves

$$\min_{\beta} \left\{ \|y - X\beta\|_2^2 \right\} \text{ subject to } \|\beta\|_2^2 \leq \kappa$$

with $\kappa = 1$.

E.3 Elastic net

Elastic net regression solves

$$\min_{\beta} \left\{ \|y - X\beta\|_2^2 \right\} \text{ subject to } \alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2 \leq \kappa$$

with $\alpha = 0.5$ and $\kappa = 1$.

E.4 Support vector regression

Support vector machine regression solves

$$\min_{w,b,\vec{\xi},\vec{\xi}^*} \frac{1}{2}\|w\|^2 + C\sum_t \xi_t + C\sum_t \xi_t^*$$

subject to

$$\begin{aligned} y_t - \vec{w}^\intercal \phi(\vec{x}_t) - b &\leq \epsilon + \xi_t^*, \\ \vec{w}^\intercal \phi(\vec{x}_t) + b - y_t &\leq \epsilon + \xi_t, \\ \xi_t^*, \xi_t &\geq 0 \quad \forall \quad t \end{aligned}$$

where $K(\vec{x}_t, \vec{x}_{t'}) = \phi(\vec{x}_t)^\intercal \phi(\vec{x}_{t'})$ is a kernel function. We use $\epsilon = 0$, $C = 800$, and choose the radial basis function as our kernel.

E.5 Artificial Neural Network

We use a multilayer perceptron that minimises the squared error loss. We use two hidden layers (thus this is a shallow neural network), tanh as our activation function, and an ℓ^2 penalty of 2000. In choosing two layers, we follow the rule of thumb that, for most problems, good performance can be achieved with just one or two hidden layers, especially if the nature of the problem tends toward being linearly separable. To solve for the weights, we use the lbfgs solver.

E.6 Random forest

We use a bootstrapped random forecast regressor with 200 trees, a max depth of 8, and a minimum sample split of 2.

F Machine learning and text models – further forecast results

This section present further results related to §5.2.

F.1 OLS AR(1) benchmark

In Table F.7 we present results from a Diebold-Mariano test for a machine learning model including text features and an AR(1) versus AR(1) OLS without the text.

F.2 ML-AR(1) benchmark

In Figure F.1 we present results for a machine learning model including text features and an AR(1) versus the same machine learning model without text. In Table F.9 we present results from a Diebold-Mariano test for the same specification.

F.3 ML-factor model and AR(1) benchmark

In Figure F.2 we present results for a machine learning model including text features, an AR(1) and factors versus the same machine learning model without text. In Table F.10 we present results from a Diebold-Mariano test for the same specification.

G Breakdown of forecast performance through time

The breakdown of differences in squared error between OLS with only an AR(1) term and OLS with text metrics and an AR(1) are shown in Figure F.3 and denoted by $\epsilon^2_{\text{Bench.}} - \epsilon^2_{\text{Text}}$. When the lines are above zero, the model with text is performing better than the model without. This shows that most of the improvement in performance comes from stressed periods.

H Variable Importance Exercise

In this section we explore the marginal effect of the features on the predicted targets following Borup et al. (2021). The approach makes use of partial dependence plots (PDP) (Friedman, 2001) and variable importance measures (Greenwell, Boehmke and McCarthy, 2018).

Let X be the time by predictors matrix that is used as inputs into a machine learning model. This includes lags of the target, macroeconomic factors, and the high dimensional set of text terms.

Suppose we are interested in examining the marginal effect of a predictor s, with vector x_s, on the expected value of the predicted target $\hat{y}_t$ for a fitted model $\hat{f}$. Letting $X_C = X \setminus x_s$, the partial dependence function for x_s is given by:

$$\text{PD}(x_s) = \hat{f}_s(x_s) = \mathbb{E}_{X_C}\left[\hat{f}(x_s, X_C)\right] = \int \hat{f}(x_s, X_C) d\mathbb{P}(X_C)$$

The x_s are the features for which the partial dependence function should be plotted and X_C are the other features used in the machine learning model $\hat{f}$ that are treated as random variables. The partial dependence function works by integrating out every variable not of interest (those in the set C) so that we are left with the relationship between the feature s and the predicted outcome. In practice, the partial dependence function is estimated via Monte Carlo samples:

$$\text{PD}(x_s) = \hat{f}_s(x_s) = \frac{1}{n}\sum_{i=1}^{T} \hat{f}(x_s, x_C^{(i)})$$

T is the number of instances in the dataset. Note that this method assumes that the features in C are not correlated with s. If this assumption is violated, the averages calculated for the partial dependence plot will include data points that are very unlikely or even impossible (see disadvantages).

To go from partial dependence to variable importance, we follow Greenwell, Boehmke and Mc-

Carthy (2018) and define

$$\mathrm{VI}(x_s) = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}\left[\hat{f}_s\left(x_s^t\right) - \frac{1}{T}\sum_{t=1}^{T}\hat{f}_s\left(x_s^t\right)\right]^2}$$

as the variable importance for numerical features. The above measure is the deviation of each unique feature value from the average curve: a flat partial dependence function indicates that a feature is not important; the more the function varies, the more important the feature is.

We take the average importance across a feature's range and, to facilitate comparison across predictors, we scale the variable importance using

$$\mathrm{VI}(x_s) = \frac{\mathrm{VI}(x_s)}{\max_k \mathrm{VI}(x_k)} \tag{1}$$

which ensures that the most important feature takes the value of unity, with other variables proportional to that.

These data are shown in Figures H.6, H.7, and H.8 for GDP, CPI, and unemployment respectively. The macro factors and lags have the highest contributions, as expected. Words that are informative for GDP include price, credit, financial, and government, while inflation sees labour, inflation, and good appear. Unemployment puts weight on house prices and income.

I Series Included in the Factor Model

Table I.11 shows the series included in the factor model. For full details of the factor model, please see Redl (2017).

J Packages and Operating System Used

J.1 Packages Used

The below lists the packages used in the preparation of this manuscript. These can be installed using the Anaconda package and environment manager. Save the information below in a file called 'mtcenv.yml' and run

```
  conda env create -f mtcenv.yml
```

on the command line.

```
name: mtcenv
channels:
  - conda-forge
dependencies:
  - autopep8
  - configparser
  - jupyterlab
  - matplotlib
  - nltk
  - numpy
  - pandas
  - pep8
  - pip
  - pylint
  - python=3.8
  - pyyaml
  - scikit-learn
  - scipy
  - seaborn
  - statsmodels
  - tqdm
  - yaml
  - fastavro
  - xlrd
  - pytest
  - black
  - arch-py
  - google-cloud-storage
  - pyarrow
  - openpyxl
  - loguru
  - rich
  - pip:
    - spacy
    - wquantiles
```

J.2 Computational Environment Used

The following Dockerfile specifies the environment used for running the code, including code on the cloud.

```
# Get Debian Linux OS docker base file
FROM continuumio/miniconda3

# Set the working directory to /app
WORKDIR /app

# Install mamba|speeds up conda
RUN conda install mamba -n base -c conda-forge

# Copy over env file
```

```
COPY mtcenv.yml /app

# create conda envt from YML file (using mamba for speed)
RUN mamba env create -f mtcenv.yml

# enable conda command in bash terminal
RUN echo ". /opt/conda/etc/profile.d/conda.sh" >> /root/.bashrc

# Make sure the Python in our newly created env is the first one on the path, so python3 picks i
ENV PATH /opt/conda/envs/mtcenv/bin:$PATH

RUN conda list

# Download required models for text packages
RUN python3 -m spacy download en_core_web_sm
RUN python3 -m nltk.downloader punkt
RUN python3 -m nltk.downloader vader_lexicon
RUN python3 -m nltk.downloader stopwords

# Copy the current directory contents into the container at /app
COPY . /app
```

Paper	Model	Target Horizon	Business Investment	CPI	Fin stab index	GDP	Hhld Consumption	IOP	IOS	Unemployment
The Daily Mail	Elastic	3	-2.65***							
		6	-2.25**							
		9	-2.22**							
	Forest	6	-1.73*			-1.97*			-1.74*	
		9	-2.46**			-1.66*	-1.98**	-1.85*		
	Lasso	6	-1.97*							
		9	-1.92*							
	NN	3	-2.98***	-2.39**		-2.65***	-2.23**		-1.66*	
		6	-2.48**	-2.88***		-1.86*	-1.88*			-2.20**
		9	-2.32**	-2.57**		-1.72*	-1.81*			-2.09**
	Ridge	3	-2.96***	-1.74*		-2.50**	-1.94*	-2.20**	-2.09**	-1.91*
		6	-2.51**	-2.14**		-1.83*	-1.67*	-1.78*		-2.25**
		9	-2.52**	-1.85*			-1.81*	-1.70*		-2.05**
	SVM	3	-2.34**	-2.18**						
		6		-2.56**						
		9		-2.43**						
The Daily Mirror	Elastic	3	-2.03**							
		6	-1.84*							
		9	-1.81*							
	Forest	6	-2.14**			-1.99**				
		9	-2.23**	-1.72*			-1.72*	-1.71*		
	Lasso	3	-2.05**							
		6	-1.78*							
		9	-1.80*							
	NN	3	-2.59**	-1.73*		-2.28**	-1.83*	-1.71*		
		6	-2.47**	-3.02***	-1.71*	-1.85*	-2.01**		-1.65*	-2.53**
		9	-2.82***	-2.79***		-1.73*				-1.80*
	Ridge	3	-2.71***	-2.20**		-2.11**		-1.83*	-1.77*	
		6	-2.63***	-1.92*		-1.67*	-1.84*			-1.90*
		9	-2.40**				-1.68*			-1.69*
	SVM	6		-1.86*						
		9		-1.94*						
The Guardian	Elastic	3	-1.94*							
		6	-1.86*							
		9	-1.72*							
	Forest	9					-1.95*			
	Lasso	3	-1.80*							
	NN	3	-3.19***	-1.74*			-1.84*			-2.14**
		6		-2.46**	-1.76*		-2.01**			-2.19**
		9		-2.78***	-1.76*		-1.82*			-2.03**
	Ridge	3	-2.05**	-2.35**						-2.49**
		6	-1.85*	-2.18**			-1.76*			-2.05**
		9		-2.20**			-1.92*			-1.88*
	SVM	6		-1.93*	-1.74*					
		9		-2.66***						

Table F.7: Results from a Diebold-Mariano test on forecasts using term frequency vectors with an AR(1) versus an AR(1) alone using OLS. Statistically significant differences in RMSE are shown. *, **, *** denote rejection of the null, of no difference in RMSE relative to the OLS AR(1), at the 10%, 5%, 1% levels respectively. Only those targets for which at least one of the machine learning models had a p-value of less than 10% are shown.

	target	Hhld Consumption	CPI	IMF fin cond	IOP	IOS	Unemployment	GDP	Fin stab index	Business Investment
horizon	type									
3	consistent	0.07*	0.04**	0.12	0.08*	0.03**	0.02**	0.05**	0.55	0.06*
	lower	0.05*	0.01***	0.10	0.08*	0.03**	0.02**	0.05**	0.09*	0.05**
	upper	0.07*	0.24	0.12	0.08*	0.03**	0.26	0.05**	0.55	0.06*
6	consistent	0.03**	0.00***	0.13	0.08*	0.05*	0.01**	0.05**	0.43	0.07*
	lower	0.03**	0.00***	0.13	0.08*	0.05*	0.01**	0.05**	0.07*	0.07*
	upper	0.03**	0.08*	0.13	0.08*	0.05*	0.02**	0.05**	0.43	0.07*
9	consistent	0.03**	0.00***	0.12	0.05*	0.05*	0.00***	0.04**	0.36	0.07*
	lower	0.03**	0.00***	0.12	0.05*	0.05**	0.00***	0.04**	0.06*	0.07*
	upper	0.03**	0.09*	0.12	0.05*	0.05*	0.00***	0.04**	0.36	0.07*

Table F.8: Results from a test for superior predictive ability (Hansen, 2005; Sheppard et al., 2021) for machine learning versus an OLS-AR(1) model. This test aims to determine whether, collectively, the alternative models outperform the benchmark model The *lower* and *upper* p-values provide a lower and upper bound for the true *p-values* respectively. Rejection of the null indicates that the competing models are significantly better that the benchmark.

		Target	Business Investment	CPI	Fin stab index	GDP	Hhld Consumption	IMF fin cond	IOP	IOS	Unemployment
Paper	Model	Horizon									
The Daily Mail	Forest	6	-1.68*							-2.17**	
		9	-2.21**			-1.67*	-2.24**				
	Lasso	9									-2.32**
	NN	3	-2.05**	-1.75*	-2.00**	-2.18**	-2.04**		-1.68*	-1.97*	
		6	-1.91*	-2.18**	-1.89*		-1.90*				
		9		-2.22**	-1.72*	-1.78*	-2.09**				
	Ridge	3		-1.91*	-1.91*	-2.18**	-2.39**	-1.77*	-2.12**	-3.38***	-1.89*
		6	-1.94*	-2.19**	-1.77*	-2.48**	-1.96*	-2.09**	-1.74*	-3.57***	-2.27**
		9	-2.15**	-2.29**		-3.15***	-2.08**	-1.69*		-4.17***	-2.50**
	SVM	3	-2.01**			-2.21**	-2.46**	-1.71*	-3.54***	-2.85***	
		6	-3.79***	-1.85*					-3.19***	-3.48***	
		9	-3.95***				-1.68*	-2.34**	-2.30**	-2.84***	
The Daily Mirror	Forest	6								-2.20**	
		9	-2.02**				-2.28**		-1.69*	-1.67*	
	NN	3	-1.99**		-2.45**	-1.79*	-1.76*		-1.71*	-2.04**	
		6		-2.20**	-1.83*		-1.87*			-1.69*	
		9	-2.11**	-2.14**	-1.86*	-1.76*	-1.95*			-1.76*	
	Ridge	3		-1.70*	-1.74*	-1.67*			-1.79*	-2.17**	-2.41**
		6	-1.75*	-2.32**	-1.72*	-2.42**	-1.97*			-2.95***	-2.52**
		9	-2.21**	-2.15**		-4.48***	-2.10**			-3.18***	-2.58**
	SVM	3	-5.67***			-2.19**	-2.23**		-3.64***	-2.86***	
		6	-3.78***	-1.87*					-2.78***	-3.45***	
		9	-4.08***					-2.44**	-2.36**	-2.52**	
The Guardian	Elastic	9					-1.95*				
	Forest	6								-2.14**	
		9				-1.72*	-2.36**		-1.90*	-1.79*	
	NN	3			-2.31**	-1.97**	-1.85*			-2.11**	-2.16**
		6		-1.80*	-2.05**	-1.72*	-1.84*			-1.76*	
		9	-2.19**	-2.20**	-1.78*	-1.83*	-2.13**		-1.98**	-1.68*	
	Ridge	3		-2.31**		-1.99**	-1.94*	-1.86*	-1.90*	-2.88***	-3.11***
		6		-2.28**		-2.60***	-1.87*	-1.74*	-1.72*	-3.79***	-2.80***
		9		-2.16**		-3.26***	-2.10**			-4.04***	-2.74***
	SVM	3				-2.24**	-2.73***		-3.54***	-2.58**	-1.72*
		6	-2.11**	-1.79*			-1.83*		-3.23***	-3.72***	
		9		-1.91*				-3.11***	-2.35**	-3.06***	-1.93*

Table F.9: Results from a Diebold-Mariano test on forecasts using term frequency vectors with an AR(1) versus an AR(1) alone with the same machine learning model. Statistically significant differences in RMSE are shown. *, **, *** denote rejection of the null, of no difference in RMSE relative to the benchmark model, at the 10%, 5%, 1% levels respectively. Only those targets for which at least one of the machine learning models had a p-value of less than 10% are shown.

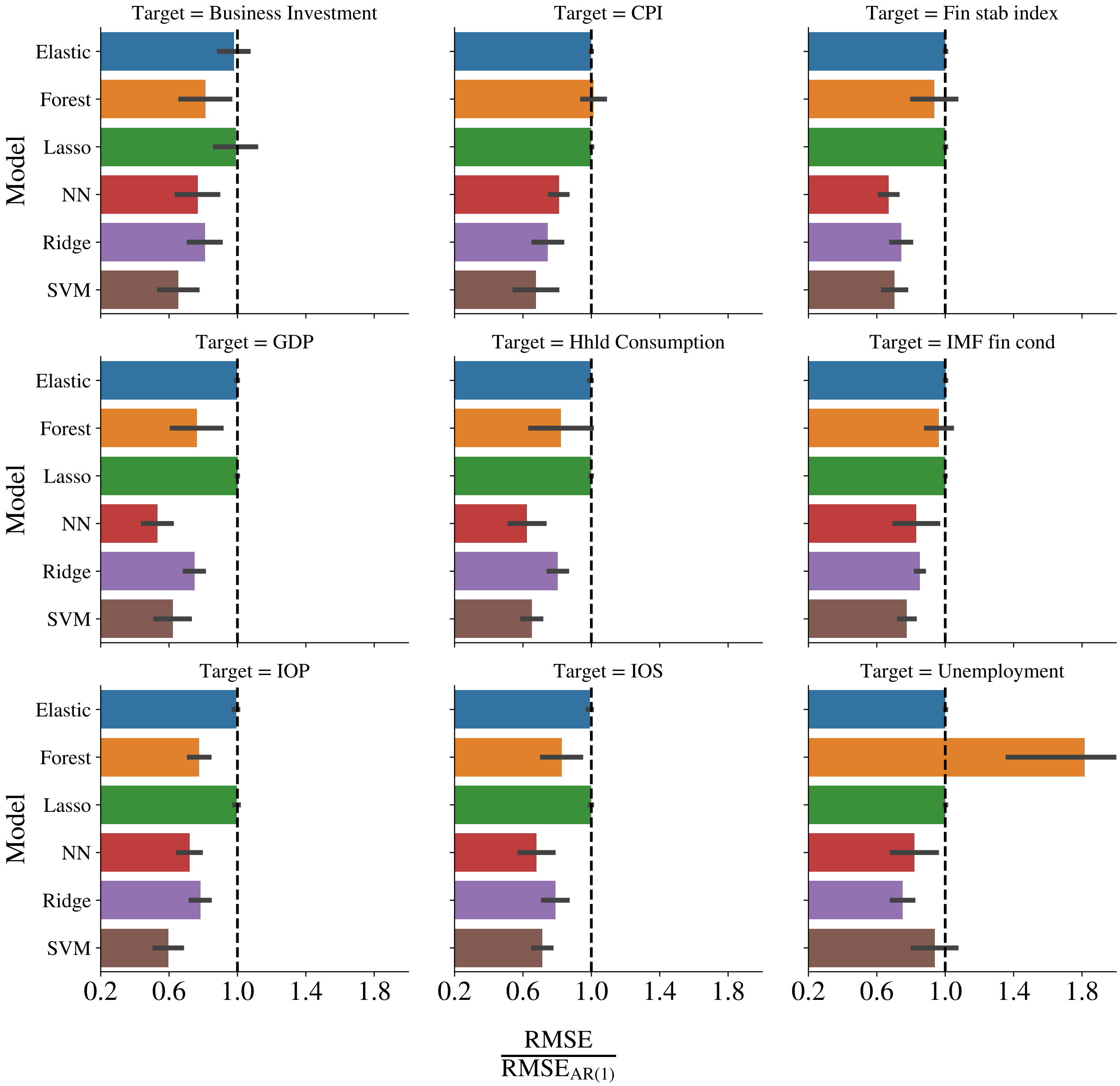

Figure F.1: RMSEs relative to a benchmark AR(1) by machine learning model and target variable. The same machine learning model (with the same hyperparameter settings) is used with text and without. Bars to the left of the dashed line indicate an improvement in forecast performance conducted with the given text metric. The confidence intervals are standard deviations over both the newspapers and the different horizons (3, 6 and 9 months ahead).

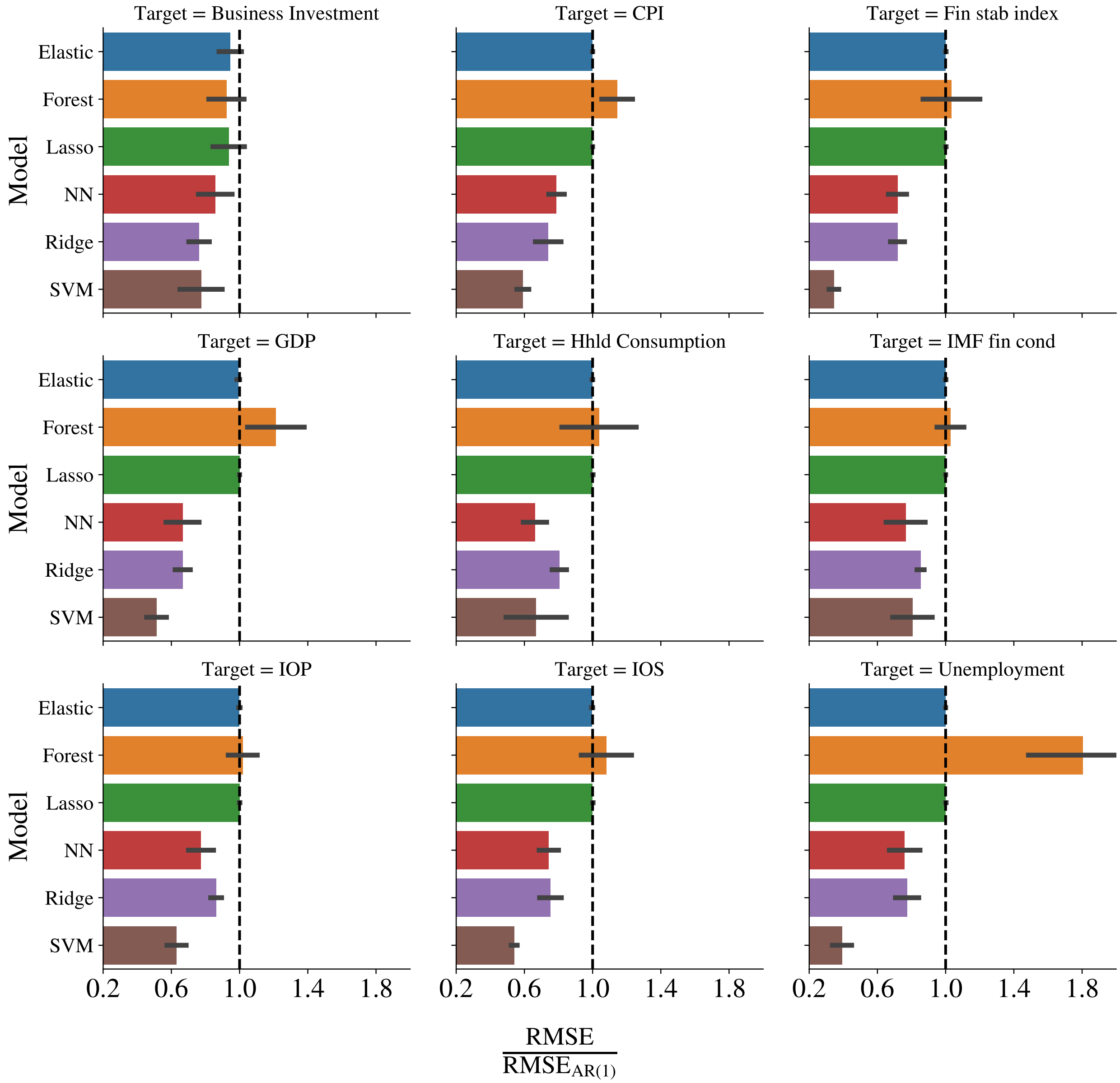

Figure F.2: The relative improvement in root mean square error of a machine learning model that uses text, an AR(1) term, and factors versus the same machine learning model with the AR(1) and factors but no text. The facets are different target variables. Bars to the left of the dashed line indicate an improvement in forecast performance conducted with the given text metric. The confidence intervals are standard deviations over both the newspapers and the different horizons (3, 6 and 9 months ahead).

		Target	Business Investment	CPI	Fin stab index	GDP	Hhld Consumption	IMF fin cond	IOP	IOS	Unemployment
Paper	Model	Horizon									
The Daily Mail	NN	3		-2.18**	-2.27**	-2.21**	-2.67***	-1.91*	-2.35**	-3.33***	-2.82***
		6		-3.06***	-1.68*	-2.81***	-3.22***		-2.00**	-3.06***	-2.63***
		9	-1.93*			-1.93*	-2.45**	-1.90*	-2.04**	-2.59**	-2.43**
	Ridge	3	-2.39**	-2.01**	-1.85*	-3.18***	-2.37**	-1.94*	-2.17**	-3.51***	-2.27**
		6	-2.15**	-2.21**		-3.11***	-2.64***	-1.85*		-2.99***	-3.77***
		9	-2.29**	-2.25**		-3.18***	-2.85***	-1.76*		-3.47***	-3.83***
	SVM	3	-3.52***	-2.91***	-3.95***	-3.13***	-4.99***	-1.76*	-3.81***	-3.42***	-5.00***
		6	-2.11**	-2.27**	-2.28**	-3.54***	-3.85***		-4.08***	-3.66***	-2.83***
		9		-3.53***	-2.26**	-3.65***			-2.56**	-2.51**	-2.22**
The Daily Mirror	Elastic	3	-1.83*	-1.70*							
	Forest	9					-2.07**				
	Lasso	6									-3.95***
	NN	3		-2.13**	-2.34**	-2.63***	-2.16**	-2.07**	-2.18**	-1.79*	-3.85***
		6	-2.26**	-2.27**	-1.90*	-2.45**	-2.26**		-2.50**	-2.25**	-2.51**
		9	-3.26***	-1.67*		-1.88*	-2.05**	-1.97*	-2.51**	-2.15**	-2.33**
	Ridge	3	-2.74***	-1.80*		-2.93***	-1.66*	-1.95*	-1.75*	-3.10***	-2.32**
		6	-2.15**	-2.47**		-2.82***	-2.14**			-3.27***	-3.30***
		9	-1.88*	-2.13**		-2.88***	-2.44**			-3.45***	-2.81***
	SVM	3	-2.55**	-2.91***	-3.90***	-3.10***	-5.54***		-3.62***	-3.06***	-4.87***
		6	-1.81*	-1.80*	-2.40**	-3.59***	-3.71***		-3.98***	-3.59***	-2.68***
		9		-3.47***	-2.31**	-4.18***			-2.29**	-2.22**	-2.08**
The Guardian	Elastic	3				-2.00**				-1.83*	
		6								-1.66*	
		9				-1.73*				-2.03**	
	NN	3	-2.09**	-1.75*	-2.23**	-1.98**	-2.69***			-2.17**	-2.35**
		6		-2.80***		-2.62***	-2.77***		-1.87*	-2.43**	-2.54**
		9		-2.31**		-2.30**	-2.76***		-1.94*	-2.17**	-2.48**
	Ridge	3		-1.89*		-3.51***	-2.10**	-2.04**	-1.84*	-3.81***	-2.83***
		6		-2.27**		-3.17***	-2.66***			-3.28***	-3.52***
		9		-2.12**		-3.06***	-2.78***			-3.23***	-3.50***
	SVM	3		-3.83***	-4.00***	-3.19***	-3.71***	-1.83*	-3.74***	-3.73***	-6.12***
		6		-2.53**	-2.27**	-3.74***	-4.03***		-4.21***	-3.95***	-3.23***
		9		-3.87***	-2.61***	-4.08***			-2.49**	-2.59**	-2.47**

Table F.10: Results from a Diebold-Mariano test on forecasts using term frequency vectors with an AR(1) and factors versus an AR(1) and factors without text using the same machine learning model. Statistically significant differences in RMSE are shown. *, **, *** denote rejection of the null, of no difference in RMSE relative to the benchmark model, at the 10%, 5%, 1% levels respectively. Only those targets for which at least one of the machine learning models had a p-value of less than 10% are shown.

Series	Source
Industrial Production	ONS
Manufacturing Production	ONS
Real Retail Sales ex Fuel	ONS
Real Retail Sales ex Food	ONS
BOP Total Exports (Goods)	ONS
Exports Volume (Goods)	ONS
BOP Total Imports (Goods)	ONS
Imports Volume (Goods)	ONS
UK CBI Survey - Below Capacity Utilisation	Thomson Reuters
CBI Industrial Trends: Current Total Order Book	Confederation of British Industry
CBI - vol of stocks bal	Confederation of British Industry
New Cars Registrations	The Society of Motor Manufacturers & Traders
LFS Unemployment Rate	ONS
LFS Number of Employees (Total)	ONS
Claimant Count Rate	ONS
LFS: total actually weekly hours worked, all	ONS
UK weekly earnings: private sector	Main Economic Indicators, OECD
PPI	ONS
CPI all items	ONS
RPI all items	ONS
RPI ex Mortgages Interest Payments (RPIX)	ONS
Nationwide House Price MoM	Housing and Construction
RICS House Price Balance	RICS - The Royal Institution of Chartered Surveyors
UK PSNCR Public Sector Net Cash Requirement	ONS
GfK Consumer Confidence	European Commission
European Commission Consumer Confidence	European Commission
CBI Distributive Trades: Retail Volume of Sales vs Year Ago	Confederation of British Industry
CBI Industrial Trends: Current Total Order Book	Confederation of British Industry
CBI Industrial Trend: Expected Selling Prices	Confederation of British Industry
Gfk/EC consumer conf, current financial situation of HH	European Commission
Gfk/EC consumer conf, current financial situation of HH over next 12m	European Commission
CBI MT expectations	Confederation of British Industry

Table I.11: The series included in the factor model.

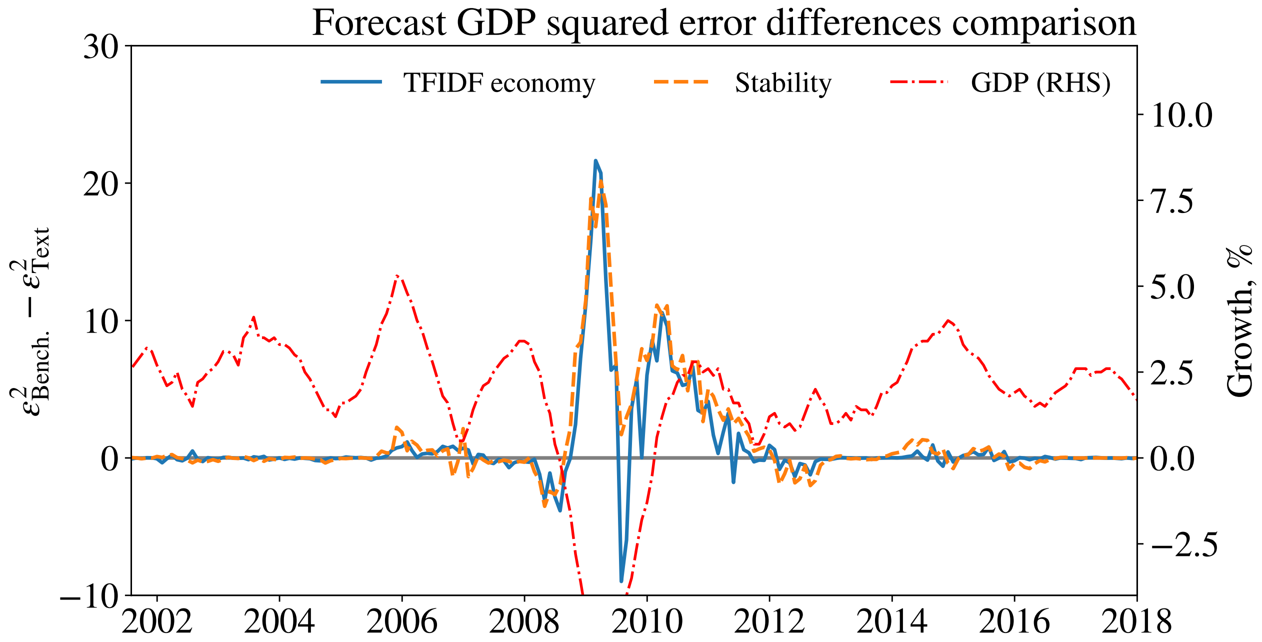

Figure F.3: Mean squared error differences between a benchmark model and text over the time-dependent union of h-month ahead out-of-sample forecasts, with horizon $h = 3, 6, 9$. The target variable is monthly GDP, shown on the right-hand axis. The benchmark is an OLS AR(1) model. The plotted error bars are standard deviations over the different horizons and newspapers. For the squared error differences, a solid line above zero means that the model with text produces smaller errors than the benchmark model. Two of the best all round performing text metrics are shown. The majority of the forecast gains are during the crisis.

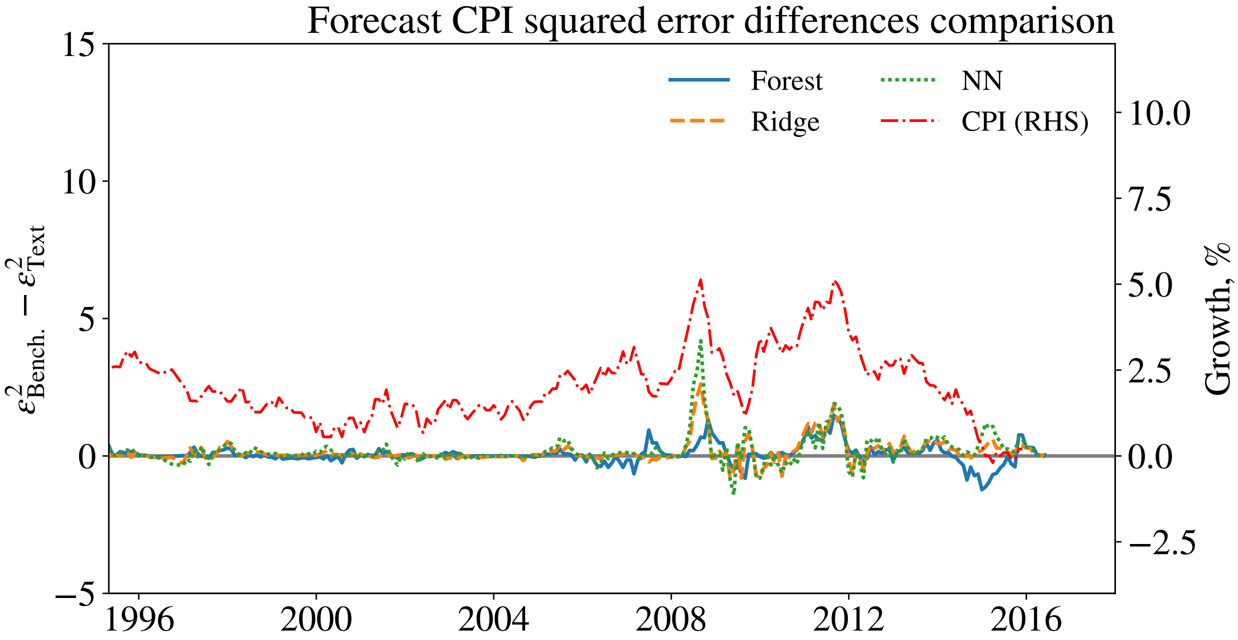

Figure F.4: Mean squared error differences between a benchmark model and text over the time-dependent union of h-month ahead out-of-sample forecasts, with horizon $h = 3, 6, 9$. The target variable is monthly inflation, shown on the right-hand axis. The benchmark is an OLS AR(1) model. The plotted error bars are standard deviations over the different horizons and newspapers. For the squared error differences, a solid line above zero means that the model with text produces smaller errors than the benchmark model. Two of the best all round performing text metrics are shown. The majority of the forecast gains are during the crisis.

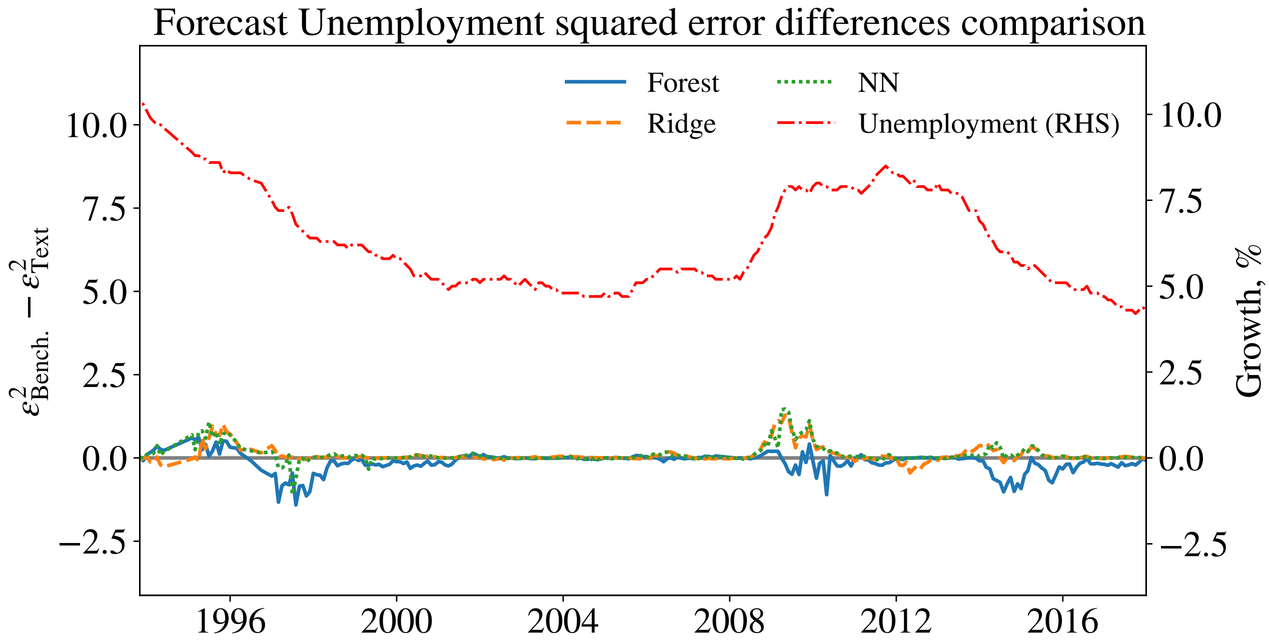

Figure G.5: Mean squared error differences between a benchmark model and text over the time-dependent union of h-month ahead out-of-sample forecasts, with horizon $h = 3, 6, 9$. The target variable is monthly inflation, shown on the right-hand axis. The benchmark is an OLS AR(1) model. The plotted error bars are standard deviations over the different horizons and newspapers. For the squared error differences, a solid line above zero means that the model with text produces smaller errors than the benchmark model. Two of the best all round performing text metrics are shown. The majority of the forecast gains are during the crisis.

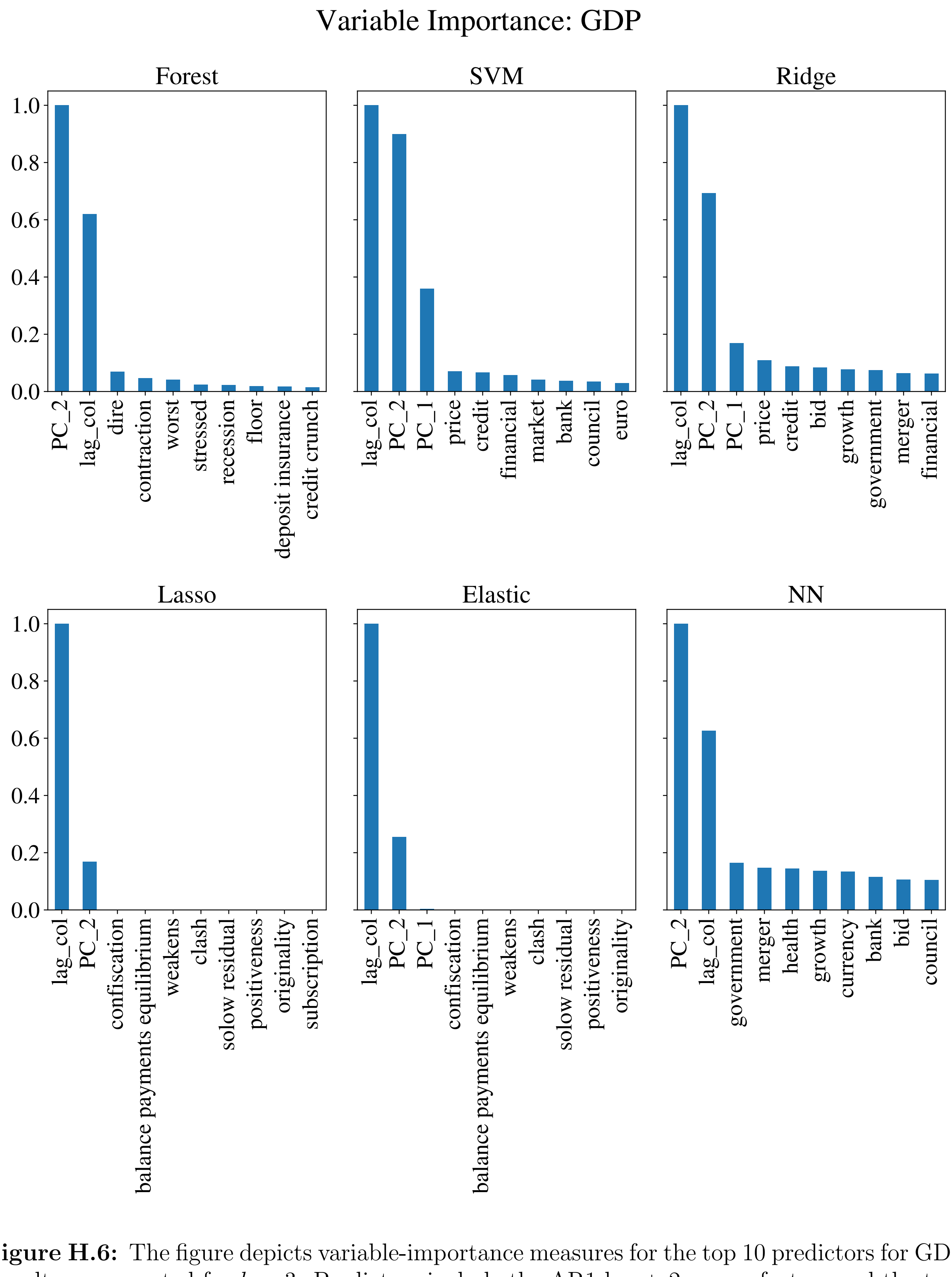

Figure H.6: The figure depicts variable-importance measures for the top 10 predictors for GDP. Results are reported for $h = 3$. Predictors include the AR1 lag + 2 macro factors and the term frequency variables. The variable importance of the most important predictor is normalised to one, and other variables are normalised relative to the most important variable.

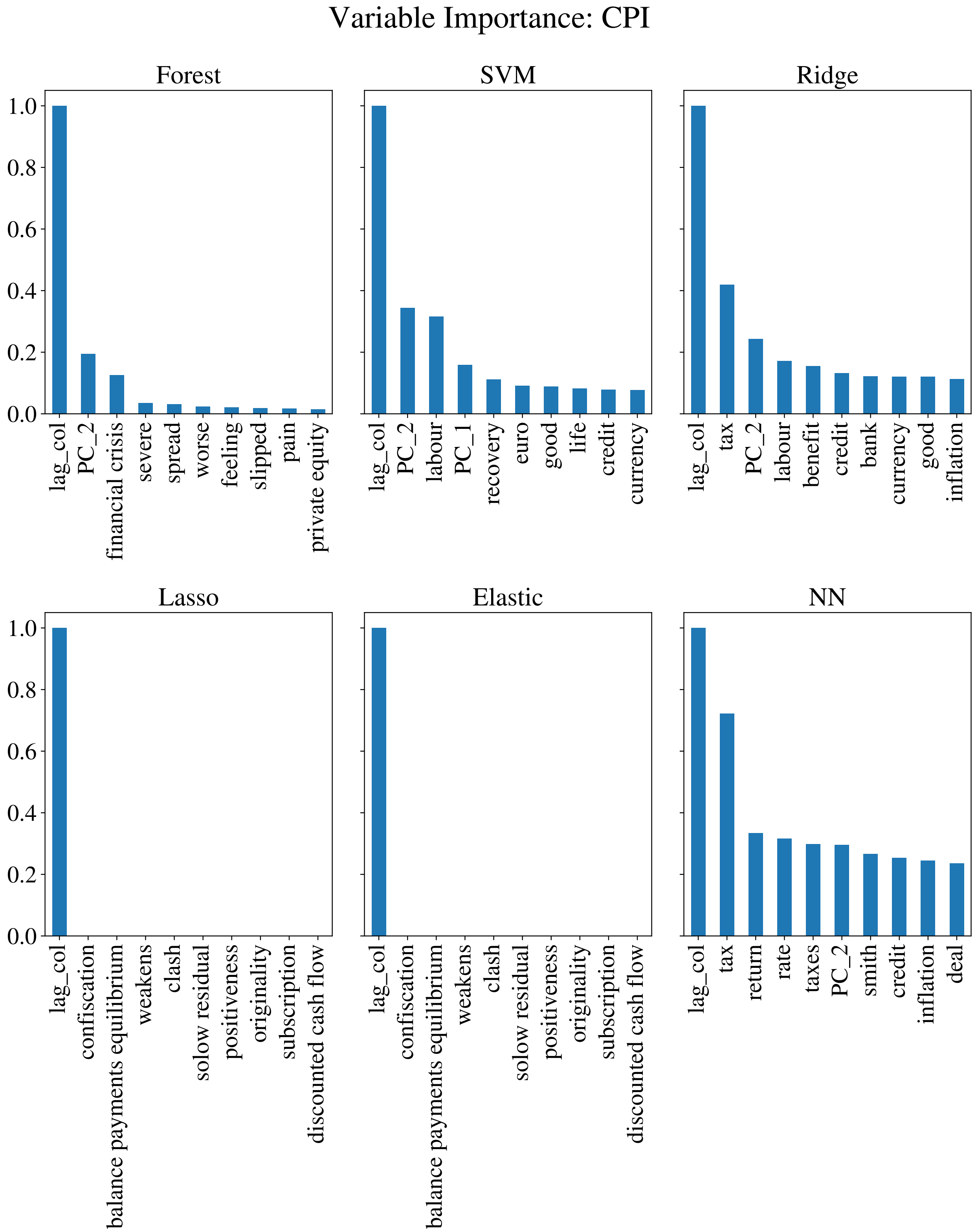

Figure H.7: The figure depicts variable-importance measures for the top 10 predictors for inflation. Results are reported for $h = 3$. Predictors include the AR1 lag + 2 macro factors and the term frequency variables. The variable importance of the most important predictor is normalised to one, and other variables are normalised relative to the most important variable.

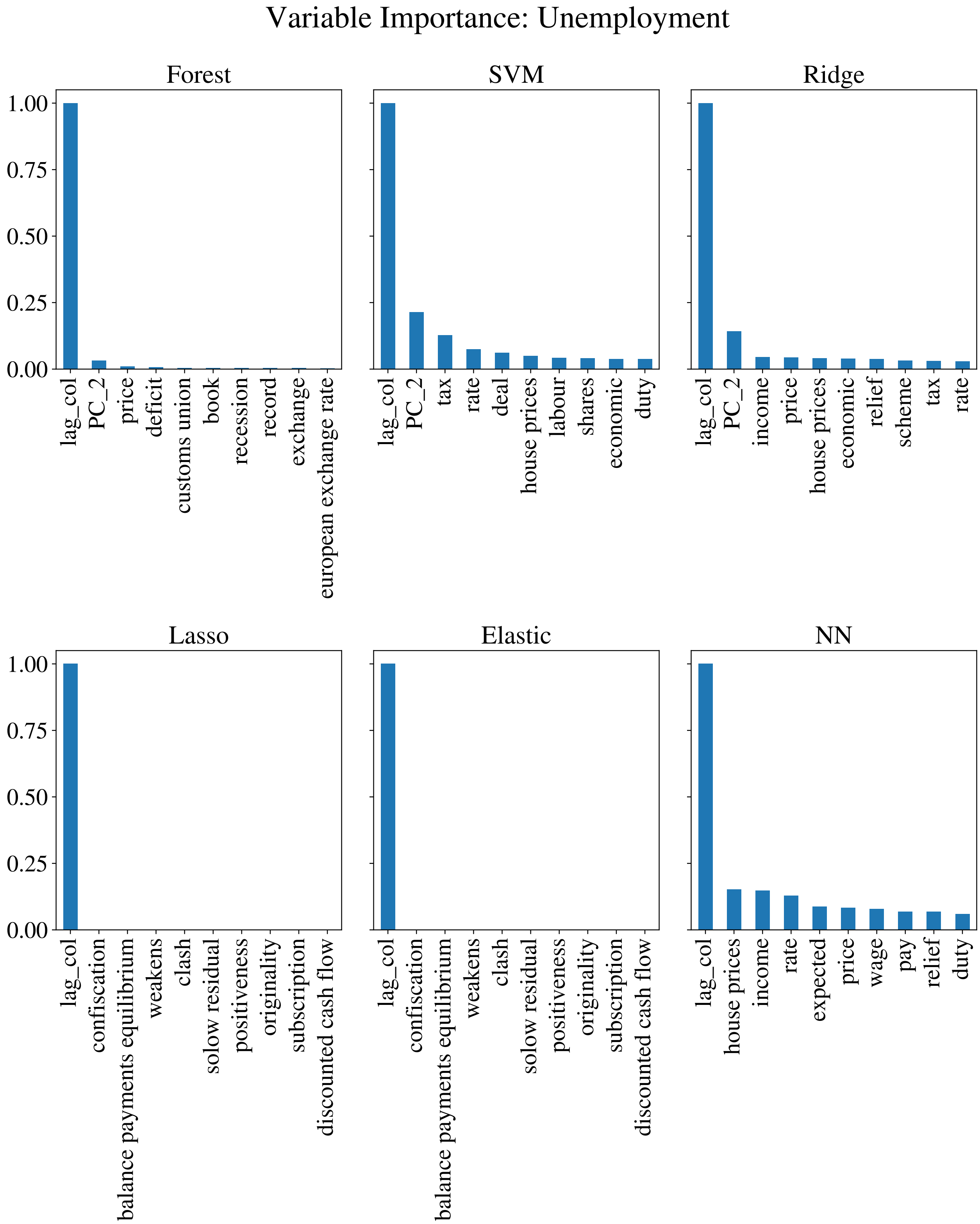

Figure H.8: The figure depicts variable-importance measures for the top 10 predictors for unemployment. Results are reported for $h = 3$. Predictors include the AR1 lag + 2 macro factors and the term frequency variables. The variable importance of the most important predictor is normalised to one, and other variables are normalised relative to the most important variable.