Finance and Economics Discussion Series

Federal Reserve Board, Washington, D.C.
ISSN 1936-2854 (Print)
ISSN 2767-3898 (Online)

Manufacturing Sentiment: Forecasting Industrial Production with Text Analysis

Tomaz Cajner, Leland D. Crane, Christopher Kurz, Norman Morin, Paul E. Soto, Betsy Vrankovich

2024-026

Please cite this paper as:
Cajner, Tomaz, Leland D. Crane, Christopher Kurz, Norman Morin, Paul E. Soto, and Betsy Vrankovich (2024). "Manufacturing Sentiment: Forecasting Industrial Production with Text Analysis," Finance and Economics Discussion Series 2024-026. Washington: Board of Governors of the Federal Reserve System, https://doi.org/10.17016/FEDS.2024.026.

Manufacturing Sentiment: Forecasting Industrial Production with Text Analysis*

Tomaz Cajner Leland D. Crane Christopher Kurz
Norman Morin Paul E. Soto Betsy Vrankovich

April 2024

Abstract

This paper examines the link between industrial production and the sentiment expressed in natural language survey responses from U.S. manufacturing firms. We compare several natural language processing (NLP) techniques for classifying sentiment, ranging from dictionary-based methods to modern deep learning methods. Using a manually labeled sample as ground truth, we find that deep learning models—partially trained on a human-labeled sample of our data—outperform other methods for classifying the sentiment of survey responses. Further, we capitalize on the panel nature of the data to train models which predict firm-level production using lagged firm-level text. This allows us to leverage a large sample of "naturally occurring" labels with no manual input. We then assess the extent to which each sentiment measure, aggregated to monthly time series, can serve as a useful statistical indicator and forecast industrial production. Our results suggest that the text responses provide information beyond the available numerical data from the same survey and improve out-of-sample forecasting; deep learning methods and the use of naturally occurring labels seem especially useful for forecasting. We also explore what drives the predictions made by the deep learning models, and find that a relatively small number of words—associated with very positive/negative sentiment—account for much of the variation in the aggregate sentiment index.

JEL codes: C1, E17, O14

Keywords: Industrial Production, Natural Language Processing, Machine Learning, Forecasting

*All authors are at the Federal Reserve Board of Governors. We thank the Institute for Supply Management, including Kristina Cahill, Tom Derry, Debbie Fogel-Monnissen, Rose Marie Goupil, Paul Lee, Susan Marty, and Denis Wolowiecki, for access to and help with the manufacturing survey data that underlie the work described by this paper. We are thankful for comments and suggestions from Stephen Hansen, Andreas Joseph, Juri Marcucci, Arthur Turrell, and participants at the Society for Government Economists Annual Conference, the ESCoE Conference on Economic Measurement, the Government Advances in Statistical Programming Conference, the Society for Economic Measurement Conference, and the Nontraditional Data, Machine Learning, and Natural Language Processing in Macroeconomics Conference. The analysis and conclusions set forth here are those of the authors and do not indicate concurrence by other members of the research staff or the Board of Governors.

1 Introduction

In recent years there has been an explosion of interest in natural language processing (NLP) within finance and macroeconomics. The use of text data to forecast and assist in model estimation is becoming increasingly commonplace. Still, there are many open questions around the use of NLP in empirical work. For example, which of the numerous available methods work best, and work best in specific contexts? Are off-the-shelf tools appropriate, or are there greater returns to specializing models to the data at hand? How useful is text for forecasting real output indicators, such as manufacturing output? What explains the predictions made by complicated NLP models? This paper addresses these questions, using a novel dataset and a variety of NLP methods ranging from traditional dictionaries to fine-tuned transformer neural networks.

Our primary data source is the monthly survey microdata underlying the Institute for Supply Management's (ISM) Manufacturing Report on Business. The survey is taken by purchasing managers at a representative sample of U.S. manufacturing firms. Part of the survey consists of categorical-response questions about aspects of their current operations, including production, inventories, backlogs, employment, and new orders. The answers to these questions are of the form "worse/the same/better than last month", and are aggregated into the widely-reported ISM diffusion indexes. But the survey also includes free-response text boxes, where purchasing managers can provide further comments either in general or about specific aspects of their businesses; these comments are a novel source of signal about the economy and our focus in this paper.[1]

Our first step is to quantify the text into an economically important and interpretable measure. We focus on sentiment, given that waves of optimism and pessimism have historically been linked to business cycle fluctuations (Keynes, 1937). We begin by evaluating various NLP methods in terms of their ability to correctly classify the sentiment expressed in individual comments. Our context is fairly specific: the data are manufacturing-sector purchasing managers opining about about the business outlook for their firm, without much discussion of financial conditions. While there are numerous sentiment classification models available, many were developed with other data in mind, such as social media posts (Nielsen, 2011). Even within economics and finance, most work has focused on finance-

[1]While ISM collects these responses through the survey, this text is confidential and not incorporated into the publicized indexes. A sample of responses are published in the monthly ISM Report on Business (see `https://www.ismworld.org/supply-management-news-and-reports/reports/ism-report-on-business/`).

related language (Araci, 2019; Correa et al., 2021; Huang et al., 2022). The lack of results for manufacturing-specific datasets motivates our assessment of a variety of NLP techniques.

One common approach is to count the frequency of words within a sentiment dictionary. Economists initially used positive and negative words from psychology literature, but have since moved on to using domain-specific words (e.g., Correa et al., 2021) and using simple word counts to measure other types of tone, such as uncertainty (see Baker et al., 2016 and Gentzkow et al., 2019). While this method is transparent, it may fail to capture negation, synonyms, and often requires context-specific dictionaries that may not be available. More recently developed techniques employ deep learning methods that account for the nuances of language. We focus on variants of BERT (see Devlin et al., 2018), a precursor of popular large language models like ChatGPT. These models are *pre-trained*: the parameters are set by exposing the model to a large corpus of text—such as the entirety of Wikipedia—and attempting to predict missing words or the relationship between sentences. The pre-trained models can be used to classify sentiment directly, or they can be further trained ("fine-tuned") on a specific dataset. The latter approach attempts to get the best of both worlds: a solid ability to parse language from the exposure to a large quantity of training data, plus the context-specific nuance from the fine-tuning data. While deep learning gets enormous attention, it is ex-ante unclear whether it should outperform carefully curated dictionaries in our context.

Comparing the accuracy of these different methods on a sample of hand-coded comments from our dataset we find that deep learning does have an advantage on our data, in part because the brevity of the comments means that many comments have no overlap with dictionary terms. In addition, we find that there is value in specializing the models to our data: the models fine-tuned on our data have the highest sentiment classification accuracy on a hold-out sample. These results point to the advantages of using pre-trained models, as well as carefully specializing them to the task at hand. Our hope is that these results help guide other economists when deciding between NLP approaches.

The sentiment measures based on free-form textual responses in the ISM data aggregate into indexes that closely mirror both the diffusion index based on the responses to the categorical survey and aggregate manufacturing output, as measured by the manufacturing component of industrial production. We further investigate the relationship between the average sentiment expressed by purchasing managers and manufacturing output econometrically. Our baseline forecasting model asks whether sentiment can help forecast manufacturing output and includes—among other controls—some of the ISM diffusion indexes, so the test

is whether the sentiment indexes have additional information beyond the ISM categorical responses data. We find that most dictionary-based text variables do not help predict manufacturing output, with the exception of a curated financial stability-specific dictionary. On the other hand, sentiment variables from the deep learning models are predictive of future manufacturing output. Out-of-sample forecasting exercises show that the financial stability dictionary and deep learning techniques significantly reduce the mean squared forecast errors as well. Overall, our results suggest that purchasing managers' survey responses contain useful forward-looking information, and that sentiment-based measures can improve the accuracy of forecasts of manufacturing output.

The exercises described above rely on a manually-labeled sample of the data, both to assess the accuracy of different methods and to help fine-tune some of the deep-learning based methods. However, the panel microdata allow for a different approach. Since firms are in the survey for multiple months, we can link the text (and other) data from a given month to next month's firm-level production data. Fitting a model to these data lets us forecast firm-level production using firm-level lagged information. This methodology has two advantages. First, it gives us a much larger training sample size as compared to the manually labeled data. Second, it aligns the training data objective very precisely with the aggregate forecasting objective. On this second point, we do our best when manually labeling data to discern whether the comment is indicative of rising or falling industrial production. But there are plenty of ambiguous cases, so there are some clear advantages to letting the data speak, and seeing what text is actually associated with future (firm level) changes in production. We find that fine-tuning in this way is competitive with using the manual labels, and in some cases preferable.

Finally, we make progress on the explainability of deep learning models. These models are notoriously opaque, a consequence of their very high parameter count and extremely nonlinear architecture. This can make it difficult to trust the outputs of such models, as it is not initially clear if the seemingly good predictions are based on solid foundations. We use a standard machine learning interpretability method—Shapley decompositions—to score the contribution of each individual word in each comment. Our results point to a sensible interpretation of our deep learning models. First, the score for each word is roughly constant over time: words do not dramatically change their average connotation (though the underlying deep learning model allows for this). Second, there are fat tails to the scores: most words have scores very close to zero (neutral), with a relatively small number of words having extreme sentiment. For example, the most positive words include

"brisk", "excellent", "booming", "improve", and "efficient"; among the most negative words are "unstable", "insufficient", "fragile", "inconsistent", and "questionable". The close-to-neutral words contribute very little to aggregate sentiment, even after accounting for the fact that they occur very frequently. Finally, we find that *changes* in our aggregated sentiment index are largely accounted for by changes in the frequency of the words with the most extreme (positive or negative) sentiment scores, with the vast majority of words playing little role. Thus, while it may be difficult to manually construct a domain-specific dictionary from scratch, it is possible to extract a fairly simple, interpretable dictionary from the deep learning model.

Our paper contributes to two strands of literature. First, our comparison of NLP techniques for measuring sentiment adds to the growing body of literature incorporating NLP into economic and financial research. Since the seminal work of Tetlock (2007), many studies have used dictionary-based methods (Baker et al., 2016; Hassan et al., 2019; Young et al., 2021; Cowhey et al., 2022), and refined lexicons for specific contexts have been shown to improve performance in measurement and forecasting (Correa et al., 2021; Gardner et al., 2022; Sharpe et al., 2023). Machine learning techniques have also been used to select word lists (Manela and Moreira, 2017; Soto, 2021). More recent papers incorporate more sophisticated machine learning methods to extract the tense and topic of texts (Angelico et al., 2022; Hanley and Hoberg, 2019; Hansen et al., 2018; Kalamara et al., 2022). Advances in NLP, particularly the use of deep learning techniques, have significantly improved sentiment classification (Heston and Sinha, 2017; Araci, 2019; Huang et al., 2022; Bybee, 2023; Jha et al., 2024).

Second, we contribute to the literature on forecasting industrial production (D'Agostino and Schnatz, 2012; Lahiri and Monokroussos, 2013; Ardia et al., 2019; Cimadomo et al., 2022; Andreou et al., 2017). Our analysis of the relationship between sentiment and industrial production provides new insights into the role of unstructured text data in economic forecasting (Marcucci, 2024). By comparing various NLP techniques, we are able to identify which methods are most effective for classifying sentiment and incorporating them into predictive models of industrial production.

The paper most similar to ours is Shapiro et al. (2022), who find that domain specific dictionaries can improve predictions of human rated sentiment. We find broadly similar results using a financial stability (rather than a general purpose) dictionary to measure sentiment, but move one step further by providing a robust comparison to large language models. Our paper differs from theirs in two important ways. First, we focus on creating

a sentiment index from firm-level data, rather than beginning the analysis at an aggregate macroeconomic level. Instead of measuring consumer sentiment through newspaper articles, we measure manufacturing sentiment from a panel of survey responses. Our unique micro-level data allow us to understand the value of text beyond categorical responses and naturally occurring labels. Second, Shapiro et al. (2022) compares lexicon-based sentiment approaches only to baseline BERT, which at the time was the most developed transfer-learning based model. We also consider newer deep learning models based on BERT, particularly those fine-tuned on domain specific and naturally occurring data. We apply interpretability techniques to these 'black box' models and show that aggregate sentiment indexes derived from deep learning hinge on the frequencies of relatively few words.

The remainder of the paper is structured as follows. Section 2 presents our data. Section 3 reviews how we measure sentiment from the textual survey data and Section 4 overviews the resulting indexes. Section 5 presents the empirical strategy and findings, and Section 6 evaluates the mechanisms through which firm survey responses predict industrial production. Section 7 concludes.

2 Data

The primary data for this study comes from the Institute for Supply Management (ISM). Each month, ISM conducts a survey of purchasing managers from a sample of manufacturing firms in the United States.[2] Diffusion indexes based on the responses (described below) are published very rapidly, and are closely watched by markets. As highlighted in Bok et al. (2018), not only does such survey data provide important signal about the state of the economy, but the ISM data in particular provides the "earliest available information for the national economy on any given quarter". In addition, the ISM data have a long time series, which is conducive to time-series modeling.[3] The timeliness and relevance of the data motivates our exploration of the free-response text.

The ISM survey includes a series of questions about the respondents' operations, including their production levels, new orders, backlog, employment, supplier delivery times, input inventories, exports, and imports. These questions have a categorical response, where the purchasing managers specify whether these metrics have increased, decreased, or stayed the same between last month and the current month. The categorical responses are aggregated

[2]ISM also surveys non-manufacturing firms and hospitals separately.

[3]ISM series extend back to 1948, but most statistical analyses use data that starts in 1972.

into publicly-released diffusion indexes, discussed more below. In addition to the categorical response, purchasing managers can provide further explanation in accompanying text boxes. There are free response questions accompanying nearly every categorical question, asking for the reason for the response. In addition there is a "General Remarks" field at the beginning, where the respondent can put any general remarks they wish. Ten to twelve of these text responses are featured in the ISM's data release to provide context for the diffusion indexes, but otherwise are not released publicly.

The ISM manufacturing survey dates back to the 1930s. The dataset we analyze covers firm-month observations from November 2001 to January 2020. Most recently, the sample covers roughly 350 responses per month. The dark-shaded area of Figure 1 shows the percentage of firms in the sample with text responses over time. The figure illustrates that the majority of respondents provide text in addition to their quantitative survey answers. The black line in Figure 1 presents the average word count over the sample period. The word counts range from 10 to 33 words on average per month. The mean word count appears to fluctuate over the business cycle and jumps dramatically in 2018. The sudden increase in word count in 2018 is mostly due to heightened tensions surrounding trade policy at the time. Indeed, after removing responses that contain the word "tariff," we observe a smoother increase in word counts (see Figure A1 in the appendix for further details).

Table 1 provides a summary of the text responses. Nearly 49 percent of the general remarks sections contain text, while the next most common sections containing text are those related to employment, production, and new orders. The last row shows statistics for all the text fields concatenated together: 69 percent of firm-month observations have any text at all, and the text is about 17 words long on average. The average word count is highest for the General Remarks section, with an average of 8 words used in these responses. When considering only those responses that contain text, the average word count for the General Remarks section increases to 16 words.

Turning from ISM's survey microdata, we use several time series in our forecasting exercises. Our focus is on forecasting the manufacturing industrial production (IP) index. We use real time data on the right hand side, reflecting what policy makers knew at the time, and forecast the fully revised series. In addition to IP series, we use the ISM diffusion indexes as regressors. The diffusion indexes are aggregations of the categorical response questions in the survey. For example, the production diffusion index is a weighted average of the responses to the production question (paraphrasing, "Is production higher/the same/lower than last month?"), with the "Higher" responses getting weight 100, "Same" responses getting weight

50, and "Lower" responses getting weight 0. The formula for the diffusion index in period t, with N_t total firms responding is shown in equation (1):

$$D_t = \frac{1}{N_t} \sum_{i=1}^{N} [100 \cdot \mathbf{1}\{\text{Response } i \text{ is "Higher"}\} + 50 \cdot \mathbf{1}\{\text{Response } i \text{ is "Same"}\}] \quad (1)$$

These diffusion indexes have values between 0 and 100, with 0 indicating that all respondents say things are worse and 100 indicating that all respondents say things are better.[4] ISM publishes indexes for each question, as well as a "PMI Composite", which is an equally-weighted average of the diffusion indexes for new orders, production, employment, supplier deliveries, and inventories.

3 Measuring Sentiment

Our goal is to extract useful information from the ISM survey text responses. We focus on sentiment analysis: measuring the extent to which the purchasing managers response is positive or negative. Even focusing on sentiment analysis, the wide range of NLP techniques available can make it challenging to choose an appropriate method. In this section we discuss the methods we use, leaving a complete description of the approaches to the Appendix.

3.1 Dictionaries

One of the simplest methods for measuring sentiment is dictionary-based analysis, which involves counting the frequency of a predetermined list of sentiment words in the text. We use common sentiment dictionaries such as the Harvard (Tetlock, 2007) and AFINN (Nielsen, 2011) word lists. However, we also recognize that certain words that may be considered negative in other contexts may not be considered negative in the context of finance, such as "taxing" or "liability". As such, we also apply finance-specific word lists, including the sentiment word list from Loughran and McDonald (2011) (henceforth, "LM") and the financial stability word list from Correa et al. (2021). For all dictionaries, we score comments on a scale of -1 to +1, using the percent of total words in the comment that are positive less the percent of total words that are negative. When we require discrete

[4]The responses are "better", "same", or "worse" for the new orders question, production, and new export orders. For employment, inventories, prices, and imports the responses are "higher", "same", and "lower". For backlogs the choices are "greater", "same", and "less".

classifications, as in Figure 2, we classify the comment as positive if the score is greater than zero, negative if it is less than zero, and neutral if it equals zero.

3.2 Deep Learning Models

Another approach to sentiment analysis involves fitting a model to the data. We try several variations on this theme. Unlike the dictionary methods, all of these approaches require labeled data: a sample of observations that have already been classified, which is used to fit the model and classify the remaining observations.

We create a labeled dataset from a randomly selected subsample of 1,000 responses *with text* from the individual questions.[5] Each response was classified for sentiment by two economists using the following question as a guide: "*Is this comment consistent with manufacturing IP rising month over month?*" The classifications were either positive, neutral, or negative, where "neutral" includes cases where is it is impossible to determine the sentiment. Both economists agreed on the sentiment classification for roughly 700 cases. This subsample is further split into a "training" dataset, used to fit the models, and "test" dataset, used to assess the relative merits of the models.[6]

Deep learning models have gained popularity in recent years, driven by their impressive performance on language-related tasks. Much of the progress has occurred within a particular class of deep learning models called *transformers* (see, e.g., Devlin et al., 2018, Radford et al., 2018, Chung et al., 2022, Ouyang et al., 2022, and Touvron et al., 2023). The defining feature of transformers—relative to other neural network architectures—is a mechanism called *attention*; a way to interact words within a sentence, allowing the context of a particular word to influence the meaning. A full explanation of transformers and the attention mechanism is beyond the scope of this paper, but we do provide a brief summary in the Appendix. The important points are that (unlike dictionaries and bag-of-words approaches) transformers take into account interactions between words, word order, and context-dependent meanings (polysemy).

One notable transformer model is "BERT", or Bidirectional Encoder Representations from Transformers, developed by Devlin et al. (2018). It is important to note that BERT is a pre-trained model: Devlin et al. (2018) specified the architecture and then trained the model on a corpus including the entirety of (English) Wikipedia and a number of books.

[5]Note, that the categorical responses can be considered a kind of label for the corresponding text. In Section 4.1 we investigate how well models can predict the categorical response from the associated text.

[6]The test data consists of observations from 2018m1 to 2020m1 and is not used by any of the models during training.

The model is large by the standards of the economics literature, with roughly 110 million parameters. We use several versions of BERT in this paper.

By default, the off-the-shelf BERT model produces sentence embeddings: Given a sentence-length piece of text, it returns a 768-dimensional vector representing the sentence. Intuitively, sentences with similar meaning ought to have embedding vectors close to each other. BERT can be used as a classifier by adding an additional layer on top of it, essentially a logistic regression that takes the embedding vector as the input and returns class probabilities. Note that this requires some labeled data to fit the logit.

BERT saves researchers the cost of training a large language model, while still allowing them to adapt the model for their specific needs, a practice known as "fine-tuning". In the financial domain, specialized BERT models have been developed to account for the unique characteristics of financial and economic text. Two prominent examples are Huang et al. (2022) (which we refer to as FinBERTv1) and Araci (2019) (which we refer to as FinBERTv2). FinBERTv1 uses the BERT architecture but is trained from scratch on SEC filings, equity reports, and earnings conference call transcripts. The sentiment classification layer is trained on the human labeled AnalystTone dataset (Huang et al., 2014).[7] FinBERTv2 was initialized with the pretrained BERT weights and further pre-trained on a corpus of Reuters news articles, which tend to focus on financial news. The sentiment classification layer was trained on the human-labeled Financial PhraseBank dataset from Malo et al. (2014).[8]

While FinBERTv1 and FinBERTv2 can do a good job parsing financial news and regulatory filings, our data are more focused on topics like order backlogs, production difficulties, inventories, and delivery times, which are not commonly found in financial corpora. After reviewing the text responses from the ISM survey, we found examples suggesting that FinBERTv1 and FinBERTv2 have some difficulty with the language. For example, the comment "slight up-tick inventory to account for slight up-tick in production" is coded as positive by the economists: it implies increased production, and an increase in input inventories to support that higher level of production. But this passage is classified as neutral by FinBERTv1 and negative by FinBERTv2. These issues motivate our use of the human-labeled dataset to fine-tune or train from scratch our own models. First, we estimate our own transformer model using the training dataset and a relatively small number of parameters. We call this

[7]Specifically, the model is `yiyanghkust/finbert-tone` from the Huggingface model hub, a classification fine-tuned version of "FinBERT-FinVocab uncased" in Huang et al. (2022).

[8]This model is `ProsusAI/finbert` on the Huggingface model hub.

model, *TF-Small* (TF for "transformer").[9] Second, we fine-tune BERT with our manually labeled training examples, and call the resulting model *Fine-Tuned BERT: Human Labeled Data*. This model benefits from the large size and extensive training of the base BERT model, but is explicitly tuned on the language relevant for our task. As we shall see below, this results in good performance.

An alternative to fine-tuning on manually-labeled data is to capitalize on the panel structure of the firm-level responses, specifically with regards to the text and the future reporting of the categorical variable measuring production. We estimate a model that uses firm f's text in month t to predict the value of firm $f's$ production in month $t+1$ (i.e. $text_{f,t}$ predicting if the firm reported $PROD_{f,t+1}$ as higher/lower/same). This strategy provides two benefits as compared to the previous approach. First, we obtain a larger dataset for fine-tuning, without having to manually label observations. Second, this approach directly aligns with our ultimate forecasting exercise, where we want to use the text at time t to predict aggregate manufacturing production. We label this BERT-based model fine-tuned on the production categorical responses as *Fine-Tuned BERT: Production Data*.[10]

Overall, we propose nine models for sentiment classification. The four dictionary-based methods are the Harvard, AFINN, Loughran and McDonald (2011), and financial stability (Correa et al., 2021) dictionaries, and the five transformer models are FinBERTv1, FinBERTv2, TF-Small, Fine-Tuned BERT: Human Labeled Data, and Fine-Tuned BERT: Production Data.

4 ISM Text-Derived Sentiment Indexes

Before evaluating the marginal value of text for forecasting aggregate series, we document the accuracy of each sentiment model on the microdata and compare our preferred sentiment indexes to the aggregate ISM composite purchasing manager index (PMI) and to US manufacturing production.

[9] We use the *Keras* library to build a simple encoder-only transformer model with input embedding dimension of 12 and an output sentiment layer with similar dimensions.

[10] The training data for Fine-Tuned BERT: Production Data includes only firm-level responses with text and for firms that appear for at least two consecutive years in the sample. The target variable for the fine-tuning is the production categorical response in month $t+1$.

4.1 Comment-Level Classification Results

Figure 2 and Table 2 present accuracy information for each model as compared the test human-labeled dataset (from 3.2).[11] The confusion matrices in Figure 2 tabulate the percent of observations with a given human "true" classification (which varies across rows) and the model-based predicted classification (which varies across columns) for each model. Overall accuracy is reported on the top of each matrix. Table 2 presents overall accuracy rates but also provides the ratio of neutral class predictions to the true number of neutral comments.

We begin by considering whether the categorical response for each comment is predictive of the human-rated sentiment. For example, if the human label for a new orders textual response is positive, we would like to know how often was the categorical response that new orders are higher than last month. We find an overall accuracy of 85.6 percent (upper left block of Figure 2), suggesting that the sentiment in the text responses—as measured by the manual label—is highly correlated with the categorical response, but not fully redundant. That is, there appears to be content in both the textual responses **and** the categorical response that might provide information related to economic activity.

The Harvard, AFINN, Loughran-McDonald, and Stability dictionaries all have accuracy scores below 30 percent (as seen in Figure 2 and Table 2.) The low accuracy is due to the fact that they predict over half of responses to be neutral, while in reality only a couple percent of responses are neutral. The last column of Table 2 shows that the dictionaries predict thirty to fifty times more neutrals than actually appear in the data. Dictionary-based methods can only produce a positive or negative classification if either positive or negative words appear in the text, and the short comments in our data often do not contain any of the words in the dictionaries.

FinBERTv1 and FinBERTv2 perform better, with accuracies of 70.3 percent and 56.8 percent, respectively. Both of these models are better able to classify actual neutral responses, but both tend to over-predict neutral classifications (though less so than the dictionaries). The best performing model is Fine-Tuned BERT: Human Labeled Data, with an accuracy score of 82.9 percent. The improved performance is largely due to having seen examples of manufacturing-specific text, as well as survey-specific examples of positive, negative, and neutral responses. The TF-Small model has an accuracy score slightly lower at 67.6 percent.

[11] While the test dataset contains 141 observations, we report predictions for only the 111 observations for which a categorical response is provided. Recall that the General Remarks response is not associated with a categorical question. This ensures the evaluation sample for the categorical response is similar to the evaluation sample for the sentiment models.

In Figure 2, we see that Fine-Tuned BERT: Production Data has the lowest performance, with an accuracy of 4.5 percent on the test dataset. This result can be attributed to a difference in the training and test samples. Nearly 55 percent of observations in the data used to train Fine-Tuned BERT: Production Data have (next month's) production flat, often with no associated text. However, our test data is derived from observations with text and include each type of question (production, new orders, etc.) as its own observation; as a result only 5 percent of the observations have the outcome of "flat"/neutral. This small share of neutral observations is also reflected in the human labels, where neutral labels are similarly rare. As a consequence, the model disproportionately labels text as "flat"/neutral in this test sample, even though it is well calibrated on the training data.

4.2 Sentiment Indexes and Aggregates

We next run the nine sentiment classifiers on all available observations, and average the sentiment scores by month. Here the scores are, in the case of the dictionaries, the fraction of positive words minus the fraction of negative words. For the transformer-based models, the scores are the predicted probability of the text being positive less the probability of the text being negative. Across all models the firm-month level scores are between -1 and 1, these are averaged by month to get aggregate sentiment.

We seasonally adjust the series using X-13 on the default settings. The seasonally adjusted time series will feed into the forecasting models in Section 5.[12] Table 3 collects the summary statistics for the monthly series. The dictionary-based monthly averages tend to have a mean close to zero and a small standard deviation, a result of the infrequent usage of words appearing in the dictionaries. In contrast, the transformers models have larger (in absolute value) means and standard deviations, largely due to the predicted probabilities that appear closer to the extremes of 1 and -1.

Table 4 shows the correlation matrix between our main variable of interest, the growth rate of manufacturing industrial production (IP Growth) as measured by the Federal Reserve Board's Industrial Production statistics,[13] and our sentiment measures. IP Growth correlates nearly 30 percent with all dictionary based sentiment measures and TF-Small, while the other deep learning based sentiment indexes exhibit stronger correlations, above 0.40. The highest correlation with IP Growth is Fine-Tuned BERT: Production Data. This might seem

[12]Note that the comment-level sentiment scores all range between -1 and +1.

[13]These data are released monthly in the Federal Reserve Board's G.17 statistical release on industrial production and capacity utilization, available at `https://www.federalreserve.gov/releases/g17/`.

puzzling given the poor performance on human labeled sentiment in Figure 2. However, our aggregate monthly sentiment measure is calculated as the difference between the percentage of positive responses less the percentage of negative responses. While this model favors neutral predictions for any given text, it labels text as positive or negative in such a way that the net sentiment (percentage positive minus percent negative) correlates highly with the best performing model for matching human sentiment, Fine-Tuned BERT: Human Labeled, as indicated by the correlation of 0.94 in Table 4. We focus on both Fine-Tuned BERT models, since one scores highly on the human labeled benchmark (Fine-Tuned BERT: Human Labeled) and the other correlates highly with IP Growth, our main variable of interest (Fine-Tuned BERT: Production Data).

Figure 3 presents a plot of the two fine-tuned BERT models and the ISM PMI aggregate. Note that the ISM PMI is on a different y-axis different axis than the net sentiment indexes. It is apparent that the sentiment indexes capture much of the dynamics of the ISM PMI. Recall that the PMI is a composite of the categorical responses, while the sentiment indexes include no direct information from the categorical responses. It is interesting that indicators based on text alone can recreate the broad dynamics of the PMI. This is reinforced by high correlations: 0.76 for the human labeled BERT model and 0.86 for the production labeled BERT model. On the other hand, it is perhaps not surprising the the series comove, as the textual responses are a supplement to the categorical answers.

The ISM manufacturing survey is by definition a report on the manufacturing sector by purchasing managers. So, it would make sense to present the text-derived sentiment measures in a figure with manufacturing industrial production. Figure 4 presents the two fine-tuned BERT models alongside the growth rate of manufacturing industrial production. Manufacturing production at the monthly frequency is quite volatile. Despite this volatility, the two fine-tuned BERT models exhibit a meaningful correlations, at 0.42 and 0.48 for human- and production-tuned BERT models respectively. We now turn to the more formal task of predicting activity with the text-derived sentiment measures.

5 Empirical Results

Our forecasting exercises focus on predicting monthly manufacturing output growth. The real time data flow is important to understand, and is as follows:

- The ISM data for a reference month t are typically released on the first business day of month $t + 1$.

- The first IP data for reference month t are typically released around the 15th of month $t+1$.

- The IP estimates for a reference month t are revised over the subsequent months and years, as more product data become available and benchmark revisions are incorporated. The monthly revisions are part of the subsequent month's IP releases, so the first monthly *revision* to IP for reference month t is released around the 15th of month $t+2$, the second revision occurs around the 15th of $t+3$, etc.

Our baseline forecasting model is as follows:

$$\Delta IP_t^{current} = \alpha + \beta_1 \Delta IP_{t-1}^{t^*} + \beta_2 \Delta IP_{t-2}^{t^*} + \beta_3 \Delta IP_{t-3}^{t^*} + \delta x_t^{t^*} + \epsilon_t \tag{2}$$

where $\Delta IP_t^{current}$ is the fully revised, current-vintage growth rate of manufacturing output in month t. The superscript t^* denotes a variable as reported on the eve of the month t G.17 IP data release: the real-time vintage relevant for forecasting ΔIP_t just prior to its first print. Thus $\Delta IP_{t-1}^{t^*}$ is the estimate of month $t-1$ from the initial month $t-1$ data release (released around the middle of month t), and $\Delta IP_{t-2}^{t^*}$ ($\Delta IP_{t-3}^{t^*}$) is the (twice) *revised* estimate of month $t-2$ ($t-3$) from the month $t-1$ data release (again, released around the middle of month t). The vector $x_t^{t^*}$ collects the ISM metrics for month t. These are available well before the month t IP data, and so may be particularly useful for forecasting. In the baseline model, x_t contains only the composite PMI index, an average of five of the ISM diffusion indexes.[14]

Table 5 presents shows in-sample results for the baseline model as well as version that add sentiment indexes. In column (1), we see that the baseline model has an R-squared of 0.219 with a positive and statistically significant relationship between PMI and IP growth. The strong relationship between PMI and IP growth illustrates the importance of the ISM categorical data as a leading indicator for production. The subsequent columns show that the aggregate sentiment indexes based on the LM, Harvard, and AFINN dictionaries are not statistically significant, and only lead to small improvements in R-squared. The only dictionary-based index that leads to a positive and significant effect on IP is the Stability dictionary, shown in column (5). The relatively good performance of the Stability dictionary is likely due to the fact that it includes several words related to the business cycle whose

[14]In unreported regressions, we forecast future industrial production, ΔIP_{t+1}, *after* IP has published for month t. The specification is: $\Delta IP_{t+1}^{current} = \alpha + \beta_1 \Delta IP_t^{t^*} + \beta_2 \Delta IP_{t-1}^{t^*} + \beta_3 \Delta IP_{t-2}^{t^*} + \delta x_t^{t^*} + \epsilon_{t+1}$. The results hold, with slightly less significance for the transformer based models in the out-of-sample exercise.

frequency would coincide with declines in the manufacturing sector, such as "contagion", "recession", and "spillover" for negative words and "healthy", "improve", and "resilient" for positive words.

Moving to columns 6-10, all five transformer-based sentiment measures are positively and significantly related to manufacturing growth. The largest gain in R-squared is seen in column (10) with Fine-Tuned BERT: Production Data. This model's fine-tuning task is to predict future firm-level (ISM-derived) production based on firm-level text, so it is reassuring that aggregate sentiment from the model predicts aggregate output.[15] The measure significantly improves on our baseline model, with a nearly 3 percentage point increase in R-squared. We also observe that the PMI index loses some of its economic and statistical significance when we include Fine-Tuned BERT: Production Data. This can be attributed to the fact that this sentiment index targets the future firm-level production in the training, overlapping with the PMI measure that aggregates firm-level production into the form of a diffusion index.

Next, we assess the out-of-sample performance of the sentiment indexes. We use two setups: the first treats the dates 2001m11-2017m12 as in-sample, and the years 2018m1-2020m1 as out-of-sample. The second focuses on the Global Financial Crisis (GFC), incorporating 2001m11-2007m11 as in-sample, and the NBER dated recession as the out-of-sample period: 2007m12-2009m6. In both cases, we respect the out-of-sample dates both when fitting the forecasting regressions and when training our upstream deep learning models. In other words, for these exercises we are only using labeled observations from the the in-sample dates (i.e. manually labeled sentiment observations from 2001m11-2017m12 to predict 2018m1-2020m1 industrial production; 2001m11-2007m11 to predict 2007m12-2009m6).

Table 6 shows the results from the expanding window exercise of forecasting IP growth over the period 2018m1-2020m1, using Diebold-Mariano tests to compare the forecast of the baseline model with text-augmented models. Each cell displays the out-of-sample RMSE and DM test statistics. In the top row—for our preferred specification—we see that the LM, Harvard and AFINN dictionary-based text measures reduce the RMSE slightly though statistically insignificantly. The Stability dictionary reduces the RSME by about 9 percent. Similarly, the transformer-based sentiment measures reduce the out-of-sample forecast errors, with FinBERTv1 and both Fine-Tuned BERT models statistically significant at the 5 percent level. The other rows in the table show alternative specifications: only including the PMI index as a control, only using lagged manufacturing growth as a control, replacing the PMI

[15]Note that the ISM data is not an input to IP, so the datasets are independent.

composite with new orders, and including several ISM diffusion indexes as controls. In nearly all cases, the Stability dictionary and transformer-based models significantly reduce the out-of-sample RMSE. In the strictest specification including three revised lags of IP growth and several ISM measures, we observe reductions in the out-of-sample RMSE of nearly 2 percent for FinBERTv1 and Fine-Tuned BERT: Human labeled Data, significant at the 1 percent level. The largest gain in forecasting is achieved by Fine-Tuned BERT: Production Data, with a reduction in RMSE of nearly 8 percent.

We extend our forecasting analysis by considering the period leading into a recession. Specifically, we rerun the out-of-sample exercise with 2001m11 to 2007m11 as the in-sample period, and using 2007m12 to 2009m7 as the out-of-sample period. Importantly, we ensure that TF-Small and the Fine-Tuned BERT models are trained only using data from 2001m11 to 2007m11. Table 7 shows the results. For dictionary-based methods, the LM and Harvard variables are slightly significant in reducing the RMSE in a few specifications. However, given that the strictest specification including lags and other ISM variables does not lead to any significant reductions in RMSE, we conclude that the dictionary-based methods do not help with forecasting during the GFC. On the other hand, we find that TF-Small and both Fine-Tuned BERT models economically and statistically improve out-of-sample forecast errors during the GFC, with a RMSE reduction in the range of 11-17 percent for the strictest specification with three lags and several ISM variables. The forecast errors for the two FinBERT models are not statistically or economically different to the baseline model. Overall, we find that sentiment variables generated from the transformer models, particularly those trained on hand-labeled and naturally occurring data, are best for improving forecasting performance during the GFC.

6 Interpretation

The results in Section 5 suggest that the sentiment indexes, and fine-tuned versions of BERT in particular, provide additional forecasting power. However, BERT is very much a black box, and it is far from obvious what drives its behavior. Machine learning models can easily make predictions based on irrelevant or unintuitive data features, an outcome we want to avoid (or at least understand). In this section we provide supporting evidence to help interpret the BERT results. We draw on the active field of research in interpretable machine learning, where many methods have been proposed to deal with these issues. We will use one such method—Shapley decompositions—to interpret our results.

Our goal is to boil down the BERT-based predictions into simple lists of relevant words with associated scores. If we can accomplish this, we will have a dictionary that is both interpretable and approximates the BERT-based models. There are several obstacles. First, it is not immediately clear how to calculate the marginal contribution of a given word to the net positive score of a comment. Second, the marginal contribution of a word can vary across comments and across time: BERT allows for context to influence the meaning of words. Finally, it is ex ante unclear whether aggregate changes in the sentiment index reflect changes in the use of many words, or a relatively small, interpretable group.

To address the first challenge we rely on Shapley decompositions (discussed below) to calculate the marginal contributions of words to comment-level scores. On the second point, we find that most words' contributions do not vary much over time, so we can treat them as approximately constant. Finally, we also find that changes in aggregate sentiment are mostly attributable to changes in the volume of extreme (positive or negative) sentiment words. Taken together, these facts suggest that the BERT-based indexes can indeed be approximated by a dictionary-based approach.

6.1 Shapley Decompositions

Shapley decompositions are used in machine learning to deal with the nonlinear relationships between the dependent variable and independent variables (Lundberg and Lee, 2017), drawing on cooperative game theory results from Shapley (1953). Given (1) an observation, and (2) the prediction of the model, the Shapley decomposition estimates the contribution of each feature to the prediction. Each contribution is relative to a "null value" for the feature; for numeric data the null value might be the mean of the feature in sample, we will discuss the null value in text data below.

Roughly speaking, the Shapley decomposition calculates the marginal contribution of switching a given feature from its null value to the observed value, averaging across all possible null/observed permutations for the *other* features. The averaging across permutations ensures that the resulting contributions have good properties, including additivity: The contributions to the prediction add up to the prediction exactly.

In our context, an observation is a single ISM comment, and the features are the individual words. BERT provides three predictions for each observation: the probability of being in the negative, neutral, and positive classes. Rather than deal with this vector, we calculate the net positive score: Pr[positive class] − Pr[negative class], and use this as the prediction. The net positive score is analogous to the diffusion index formula, and reduces the model

output to a single number between -1 and +1.

To understand how the Shapley decomposition operates in our context, consider the example comment "`Business continues to be slow`". Fine-tuned BERT predicts this comment is positive with probability 0.078, with a net positive probability of -0.76. The Shapley decomposition replaces subsets of the words with a special token, `[MASK]`.[16] BERT interprets `[MASK]` as meaning that there is a real, unknown word in that place in the comment. BERT continues to make predictions for the class of the comment even when words are masked; these predictions are based on the remaining unmasked words and the positions of the words in the comment.

The marginal contribution of the word "`slow`" can be calculated as the difference between the net positive probability of "`Business continues to be slow`" and "`Business continues to be [MASK]`". However another plausible estimate of the marginal contribution would be, e.g., the difference between "`[MASK] continues to be slow`" and "`[MASK] continues to be [MASK]`". The Shapley decomposition iterates over the various masking permutations to arrive at an average marginal contribution.[17]

It is worth noting here that the Shapley decomposition is not a structural explanation, nor does it imply any causal relationship. It is an accounting identity that can be imposed on any model. For our purposes, it is useful for linearizing the the relationship between tokens and the aggregate sentiment index.

After running the Shapley decomposition on all the comments, we obtain‘ *Shapley scores* for each token in each comment. The Shapley scores for the tokens in a given comment add up to the net positive probability for that comment. The contribution of a token can vary across comments, because BERT's predictions are not a linear function of the tokens. This is part of the advantage of BERT: tokens may have different meanings depending on the context. However, in order to get a handle on interpretability we average that variation away and work with time-invariant word-level Shapley scores.

We can also examine the distribution of words across scores: Figure 5 plots the density of words across Shapley scores. The density is winsorized at the top and bottom 5 percent of the distribution to make the central mass visible. The weighted density, in black, shows the distribution weighted by the number of occurrences in the corpus. The vermilion unweighted density counts each unique token in the vocabulary equally. Note that many tokens have

[16] In NLP, "tokens" are the basic unit of observation, they can be words or word parts.

[17] In practice, calculating every permutation requires 2^N model evaluations for a sequence with N tokens, which can become very costly even for short comments. The SHAP package for Python circumvents this issue by sampling.

scores close to zero, particularly in the weighted plot. As the Shapley scores can range between positive and negative 1, it might be puzzling why so much mass is concentrated on the (-0.01,0.01) interval. Part of the reason is simply the length of the comments: if comments are on average 16 words long, a random word will—on average—only contribute 1/16th to the comment's score (which is bounded on (-1,1)). In addition, many of the tokens are filler words or word parts, e.g., the token "the" has a Shapley score of 0.003.

6.2 Approximate Sentiment Index

Next, we make two approximations. First, we replace the Shapley scores for each word in each comment with that word's average Shapley score across all time. This amounts to imposing that each word has a time-invariant sentiment score.

Second, motivated by the aforementioned histograms, we focus only on the tails of the distribution. In particular we keep only the top and bottom 5 percent of words, on the theory that they contain the most information about sentiment. This restriction reduces the number of words in the vocabulary to about 1,000 (from more than 10,000).

We recalculated an approximate sentiment index by adding up the (time averaged) word-level Shapley scores for the top and bottom 5 percent of words, then dividing by the number of comments. Figure 6 shows the results. The approximate sentiment index is closely aligned with the standard index, although it does not fall by as much during the Great Recession. In general, it appears that the much simpler approximate sentiment index captures the important features of the original. This is useful, given that the approximate sentiment index is essentially a dictionary based method: it is constructed by adding up the (time-averaged) word-level sentiment scores of each word. Table 8 shows the words with the most positive and negative sentiment scores. The words in each group appear quite reasonable, which reassures us that BERT is indeed picking up on meaningful semantics in the comments.

7 Conclusion

In this paper, we examine the relationship between manufacturing sentiment and industrial production growth, an important indicator for macroeconomic forecasting. To evaluate the effectiveness of the sentiment measures, we compare dictionary-based and deep learning methods to human labeled sentiment scores. Our results show that context-specific dictionary-based methods and deep learning techniques perform best in mimicking human sentiment classifications for individual comments. Notably, indexes based on the average sen-

timent measures of free-form textual responses closely mirror the ISM diffusion index and manufacturing production. In addition, when estimating out-of-sample industrial production growth, we find that sentiment measures based on financial-stability focused words and fine-tuned deep learning models significantly improve forecasting accuracy. Our preferred deep learning model, one based on naturally occurring labels, is consistent with a view that most words have nearly-neutral sentiment, with aggregate sentiment and changes in aggregate sentiment hinging on the frequencies of relatively few words with extreme sentiment scores.

Our comparison of different sentiment measures can assist future researchers in choosing the most appropriate methodology for text analysis. Our findings suggest that deep learning techniques benefit from both manual and naturally occurring labels, and that context-specific dictionaries outperform general purpose dictionaries in out-of-sample exercises. With the advent of large language models, such as ChatGPT, we hope future research can test whether fine-tuning or curating dictionaries are needed with generative artificial intelligence. Furthermore, the improvements to industrial production forecasts we find using survey responses suggest that other macroeconomic variables may also benefit from the inclusion of unstructured and non-traditional data such as text.

References

Andreou, Elena, Patrick Gagliardini, Eric Ghysels, and Mirco Rubin, "Is Industrial Production Still the Dominant Factor for the US Economy?," CEPR Discussion Papers 12219 August 2017.

Angelico, Cristina, Juri Marcucci, Marcello Miccoli, and Filippo Quarta, "Can We Measure Inflation Expectations Using Twitter?," *Journal of Econometrics*, 2022, *228* (2), 259–277.

Araci, Dogu, "FinBERT: Financial Sentiment Analysis with Pre-trained Language Models," *arXiv preprint arXiv:1908.10063*, 2019.

Ardia, David, Keven Bluteau, and Kris Boudt, "Questioning the News About Economic Growth: Sparse Forecasting Using Thousands of News-Based Sentiment Values," *International Journal of Forecasting*, 2019, *35* (4), 1370–1386.

Baker, Scott R., Nicholas Bloom, and Steven J. Davis, "Measuring Economic Policy Uncertainty," *The Quarterly Journal of Economics*, 2016, *131* (4), 1593–1636.

Bok, Brandyn, Daniele Caratelli, Domenico Giannone, Argia M. Sbordone, and Andrea Tambalotti, "Macroeconomic Nowcasting and Forecasting with Big Data," *Annual Review of Economics*, 2018, *10* (1), 615–643.

Bybee, J Leland, "The Ghost in the Machine: Generating Beliefs with Large Language Models," *arXiv preprint arXiv:2305.02823*, 2023.

Chung, Hyung Won, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei, "Scaling Instruction-Finetuned Language Models," *arXiv preprint arXiv:2210.11416*, 2022.

Cimadomo, Jacopo, Domenico Giannone, Michele Lenza, Francesca Monti, and Andrej Sokol, "Nowcasting with Large Bayesian Vector Autoregressions," *Journal of Econometrics*, 2022, *231* (2), 500–519.

Correa, Ricardo, Keshav Garud, Juan M. Londono, and Nathan Mislang, "Sentiment in Central Banks' Financial Stability Reports," *Review of Finance*, 2021, *25* (1), 85–120.

Cowhey, Maureen, Seung Jung Lee, Thomas Popeck Spiller, and Cindy M. Vojtech, "Sentiment in Bank Examination Reports and Bank Outcomes," FEDS Working Paper, Board of Governors of the Federal Reserve System 2022.

D'Agostino, Antonello and Bernd Schnatz, "Survey-based nowcasting of US growth: a real-time forecast comparison over more than 40 years," Working Paper Series 1455, European Central Bank August 2012.

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

Gardner, Ben, Chiara Scotti, and Clara Vega, "Words speak as loudly as actions: Central bank communication and the response of equity prices to macroeconomic announcements," *Journal of Econometrics*, 2022, *231* (2), 387–409.

Gentzkow, Matthew, Bryan Kelly, and Matt Taddy, "Text as Data," *Journal of Economic Literature*, 2019, *57* (3), 535–74.

Hanley, Kathleen Weiss and Gerard Hoberg, "Dynamic Interpretation of Emerging Risks in the Financial Sector," *The Review of Financial Studies*, 2019, *32* (12), 4543–4603.

Hansen, Stephen, Michael McMahon, and Andrea Prat, "Transparency and Deliberation Within the FOMC: A Computational Linguistics Approach," *The Quarterly Journal of Economics*, 2018, *133* (2), 801–870.

Hassan, Tarek A., Stephan Hollander, Laurence Van Lent, and Ahmed Tahoun, "Firm-Level Political Risk: Measurement and Effects," *The Quarterly Journal of Economics*, 2019, *134* (4), 2135–2202.

Heston, Steven L. and Nitish Ranjan Sinha, "News vs. Sentiment: Predicting Stock Returns from News Stories," *Financial Analysts Journal*, 2017, *73* (3), 67–83.

Huang, Allen H, Amy Y Zang, and Rong Zheng, "Evidence on the Information Content of Text in Analyst Reports," *The Accounting Review*, 2014, *89* (6), 2151–2180.

Huang, Allen H., Hui Wang, and Yi Yang, "FinBERT: A Large Language Model for Extracting Information from Financial Text," *Contemporary Accounting Research*, 2022, *40* (2), 806–841.

Jha, Manish, Jialin Qian, Michael Weber, and Baozhong Yang, "ChatGPT and Corporate Policies," Technical Report, National Bureau of Economic Research 2024.

Kalamara, Eleni, Arthur Turrell, Chris Redl, George Kapetanios, and Sujit Kapadia, "Making Text Count: Economic Forecasting Using Newspaper Text," *Journal of Applied Econometrics*, 2022, *37* (5), 896–919.

Keynes, John Maynard, "The General Theory of Employment," *The Quarterly Journal of Economics*, 1937, *51* (2), 209–223.

Lahiri, Kajal and George Monokroussos, "Nowcasting US GDP: The Role of ISM Business Surveys," *International Journal of Forecasting*, 2013, *29* (4), 644–658.

Loughran, Tim and Bill McDonald, "When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks," *The Journal of Finance*, 2011, *66* (1), 35–65.

Lundberg, Scott M. and Su-In Lee, "A Unified Approach to Interpreting Model Predictions," *arXiv preprint arXiv:1705.07874*, 2017.

Malo, Pekka, Ankur Sinha, Pekka Korhonen, Jyrki Wallenius, and Pyry Takala, "Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts," *Journal of the Association for Information Science and Technology*, 2014, *65* (4), 782–796.

Manela, Asaf and Alan Moreira, "News Implied Volatility and Disaster Concerns," *Journal of Financial Economics*, 2017, *123* (1), 137–162.

Marcucci, Juri, "Macroeconomic Forecasting with Text-Based Data," Working paper, Bank of Italy 2024.

Nielsen, Finn Årup, "A New ANEW: Evaluation of a Word List for Sentiment Analysis in Microblogs," *arXiv preprint arXiv:1103.2903*, 2011.

Ouyang, Long, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe,

"Training Language Models to Follow Instructions with Human Feedback," *arXiv preprint arXiv:2203.02155*, 2022.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever, "Language Models are Unsupervised Multitask Learners," Technical Report 2018.

Shapiro, Adam Hale, Moritz Sudhof, and Daniel J. Wilson, "Measuring News Sentiment," *Journal of Econometrics*, 2022, *228* (2), 221–243.

Shapley, L. S., "A Value for n-Person Games," in Harold William Kuhn and Albert William Tucker, eds., *Contributions to the Theory of Games (AM-28), Volume II*, Princeton: Princeton University Press, 1953, pp. 307–318.

Sharpe, Steven A., Nitish R. Sinha, and Christopher A. Hollrah, "The Power of Narrative Sentiment in Economic Forecasts," *International Journal of Forecasting*, 2023, *39* (3), 1097–1121.

Soto, Paul E., "Breaking the Word Bank: Measurement and Effects of Bank Level Uncertainty," *Journal of Financial Services Research*, 2021, *59* (1), 1–45.

Tetlock, Paul C., "Giving Content to Investor Sentiment: The Role of Media in the Stock Market," *The Journal of Finance*, 2007, *62* (3), 1139–1168.

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample, "LLaMA: Open and Efficient Foundation Language Models," *arXiv preprint arXiv:2302.13971*, 2023.

Young, Henry L., Anderson Monken, Flora Haberkorn, and Eva Van Leemput, "Effects of Supply Chain Bottlenecks on Prices using Textual Analysis," FEDS Notes, Board of Governors of the Federal Reserve System 2021.

Tables

Table 1: Survey Summary Statistics

Field	(1) Fraction W/ Text	(2) Mean Word Count	(3) Mean Word Count Cond. on Text
General Remarks	0.49	7.86	15.94
Production	0.27	1.42	5.27
New Orders	0.27	1.45	5.41
Backlog	0.19	1.13	6.05
Employment	0.52	1.29	6.21
Supplier Speed	0.12	0.89	7.23
Input Inventories	0.23	1.51	6.50
Exports	0.10	0.59	5.88
Imports	0.12	0.79	6.43
All Text (Appended)	0.69	16.92	24.42

Notes: Summary statistics derived from the ISM survey. Column (1) reports the fraction of firm-month observations containing any text. Column (2) shows the mean word count across all firm-month observations, while column (3) shows the mean word count of only those responses containing any text. Each row corresponds to one of the various question types on the ISM survey.

Table 2: Sentiment Classification Accuracy

Model	Accuracy	Predicted neutrals, relative to actual
AFINN	27.93	37.00
LM	20.72	43.50
Harvard	24.32	37.50
Stability	11.71	50.00
FinBERT (v1)	70.27	13.00
FinBERT (v2)	56.76	17.50
TF-Small	67.57	1.50
Fine-Tuned BERT: Human Labeled Data	82.88	0.00
Fine-Tuned BERT: Production Data	4.50	54.00

Notes: Accuracy and other statistics for sentiment classification models. All evaluations are done on the hold-out data. "Accuracy" is the percent of observations correctly classified by the model. The third column shows the ratio of neutral class predictions to the true number of neutral comments.

Table 3: Summary Statistics

N=219	Mean	Std. Dev.	Min	Median	Max
Text Measures					
LM	-0.0096	0.0064	-0.0399	-0.0097	0.0046
Harvard	0.0005	0.0048	-0.0216	0.0009	0.0138
AFINN	0.0123	0.0109	-0.0300	0.0115	0.0374
Stability	-0.0012	0.0040	-0.0233	-0.0012	0.0095
FinBERT (v1)	-0.0379	0.1111	-0.4454	-0.0313	0.2019
FinBERT (v2)	-0.0633	0.1024	-0.4882	-0.0442	0.1561
TF-Small	0.1864	0.1403	-0.2559	0.1954	0.9813
Fine-Tuned BERT: Human Labeled Data	0.1082	0.0810	-0.2261	0.1156	0.3141
Fine-Tuned BERT: Production Data	0.1100	0.0310	-0.0162	0.1128	0.1733
Macro Variables					
IP Growth$_t$	0.0335	0.7041	-3.4210	0.0406	1.5950
ISM_PMI$_t$	53.0959	4.6551	34.5000	53.2000	61.4000
ISM_NewOrders$_t$	55.8511	6.6517	25.9000	56.6000	71.3000
ISM_Inventories$_t$	47.9950	4.2814	33.5000	48.6000	56.8000

Notes: Summary statistics for the variables used in the in-sample and out-of-sample analysis. LM, Harvard, AFINN, and Stability measure the average net sentiment when applying dictionary word counts of the Loughran and McDonald (2011), Harvard, AFINN, and Stability (Correa et al., 2021) word lists, respectively. FinBERT (v1) and FinBERT (v2) measure the average net sentiment of applying the FinBERT model from Huang et al. (2022) and Araci (2019), respectively. TF-Small and Fine-Tuned BERT: Human Labeled Data are sentiment scores derived from a fine-tuned transformer and a fine-tuned BERT model using a sample of human-labeled ISM responses. Fine-Tuned BERT: Production Data is a fine-tuned BERT model using panel data of the firm-level responses to predict the categorical variable for production in $t+1$ using the text in month t.

Table 4: Correlation Matrix

	IP Growth	LM	Harvard	AFINN	Stability	FinBERT (v1)	FinBERT (v2)	TF-Small	FT BERT: Human Labeled Data	FT BERT: Production Data
IP Growth	1.000									
LM	0.308	1.000								
Harvard	0.252	0.499	1.000							
AFINN	0.247	0.699	0.534	1.000						
Stability	0.243	0.664	0.375	0.641	1.000					
FinBERT (v1)	0.437	0.607	0.492	0.576	0.384	1.000				
FinBERT (v2)	0.443	0.641	0.523	0.596	0.412	0.915	1.000			
TF-Small	0.315	0.519	0.459	0.450	0.272	0.695	0.786	1.000		
FT BERT: Human Labeled Data	0.423	0.631	0.534	0.537	0.346	0.907	0.938	0.819	1.000	
FT BERT: Production Data	0.483	0.609	0.514	0.502	0.345	0.910	0.927	0.707	0.938	1.000

Notes: Correlation matrix between manufacturing industrial production growth (IP Growth) and our nine sentiment measures. LM, Harvard, AFINN, and Stability measure the average net sentiment when applying dictionary word counts of the Loughran and McDonald (2011), Harvard, AFINN, and Stability (Correa et al., 2021) word lists, respectively. FinBERT (v1) and FinBERT (v2) measure the average net sentiment of applying the FinBERT model from Huang et al. (2022) and Araci (2019), respectively. TF-Small and Fine-Tuned BERT: Human Labeled Data are sentiment scores derived from a fine-tuned transformer and a fine-tuned BERT model using a sample of human-labeled ISM responses. Fine-Tuned BERT: Production Data is a fine-tuned BERT model using panel data of the firm-level responses to predict the categorical variable for production in $t+1$ using the text in month t.

Table 5: In-sample Regression Results

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
	Dependent Variable: IP Growth$_t$									
		Dictionary Based Methods				*Deep Learning Methods*				
Text Measure	*Baseline*	LM	Harvard	AFINN	Stability	FinBERT (v1)	FinBERT (v2)	TF-Small	Fine-Tuned BERT: Human Labeled Data	Fine-Tuned BERT: Production Data
ISM_Sentiment$_t$		0.0801	0.0475	0.0349	0.114***	0.142*	0.164**	0.0909*	0.118*	0.227***
		(0.0500)	(0.0548)	(0.0486)	(0.0410)	(0.0730)	(0.0695)	(0.0493)	(0.0626)	(0.0831)
ISM_PMI$_t$	0.0660***	0.0609***	0.0633***	0.0648***	0.0672***	0.0499***	0.0496***	0.0615***	0.0519***	0.0332**
	(0.0147)	(0.0139)	(0.0145)	(0.0144)	(0.0148)	(0.0152)	(0.0133)	(0.0139)	(0.0141)	(0.0160)
IP Growth$_{t-1}$	-0.0303	-0.0473	-0.0422	-0.0391	-0.0486	-0.0555	-0.0555	-0.0401	-0.0493	-0.0515
	(0.0908)	(0.0883)	(0.0900)	(0.0906)	(0.0867)	(0.0894)	(0.0883)	(0.0897)	(0.0883)	(0.0860)
IP Growth$_{t-2}$	0.0611	0.0447	0.0511	0.0525	0.0249	0.0232	0.0127	0.0417	0.0366	0.0224
	(0.0947)	(0.0907)	(0.0923)	(0.0935)	(0.0875)	(0.0953)	(0.0951)	(0.0928)	(0.0926)	(0.0921)
IP Growth$_{t-3}$	0.0248	0.00706	0.0187	0.0166	0.00158	-0.00151	-0.0198	-0.00274	0.00484	-0.0173
	(0.0963)	(0.0953)	(0.0954)	(0.0963)	(0.0949)	(0.0991)	(0.0977)	(0.0956)	(0.0955)	(0.0981)
Observations	219	219	219	219	219	219	219	219	219	219
R-squared	0.219	0.228	0.222	0.221	0.243	0.234	0.241	0.231	0.230	0.248

Notes: This table reports in-sample regressions of the month-to-month percentage change of industrial production on a set of real-time predictors of IP from 2001m11-2020m1. $ISM\_Sentiment$ is a text measure of the survey response sentiment using either dictionary-based methods (columns 2-5), transfer learning of financial BERT models (columns 6-7), or fine-tuned models trained on a random selection of human-labeled ISM survey responses (columns 8-9) or naturally occurring labels predicting future production (column 10). We standard normalize the sentiment measures in the regressions so that the coefficient can be interpreted as the increase in the change of industrial production in response to a one standard deviation increase in sentiment. ISM_PMI is the monthly diffusion index of PMI released by the ISM at the beginning of the month. IP_Growth_{t-1} is the estimate of month $t-1$ from the initial month $t-1$ data release, and IP_Growth_{t-2} (IP_Growth_{t-3}) is the (twice) *revised* estimate of month $t-2$ ($t-3$) from the month $t-1$ data release. Significance levels are indicated by *** (1 percent), ** (5 percent), and * (10 percent).

Table 6: Out-of-sample Regression Results (2018m1-2020m1)

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)		
In-Sample: 2001M11-2017M12 Out-of-Sample: 2018M1-2020M1												
		Dictionary Based Methods				*Deep Learning Methods*					Controls Included	
Text Measure	*Baseline*	LM	Harvard	AFINN	Stability	FinBERT (v1)	FinBERT (v2)	TF-Small	Fine-Tuned BERT: Human Labeled Data	Fine-Tuned BERT: Production Data	IP Lags	ISM Variables
OOS MSE	0.444	-2.93%	-0.45%	-0.45%	-9.23%	-6.53%	-11.71%	-3.60%	-4.95%	-15.09%	3 Lags	PMI
DM Test P-Value		0.271	0.572	0.845	0.178	0.033	0.073	0.425	0.03	0.016		
OOS MSE	0.446	-3.81%	-1.35%	-1.12%	-10.31%	-8.07%	-13.00%	-4.93%	-6.28%	-16.14%	-	PMI
DM Test P-Value		0.161	0.319	0.683	0.14	0.027	0.046	0.249	0.013	0.011		
OOS MSE	0.391	-2.30%	-1.53%	1.28%	2.56%	-5.63%	-7.16%	-3.32%	-7.67%	-16.62%	3 Lags	-
DM Test P-Value		0.354	0.576	0.614	0.637	0.111	0.087	0.243	0.075	0.018		
OOS MSE	0.425	-0.94%	-0.47%	0.71%	-3.76%	-1.41%	-4.24%	-0.94%	-1.41%	-6.35%	3 Lags	New Orders
DM Test P-Value		0.117	0.599	0.087	0.112	0.002	0.057	0.767	0.03	0.001		
OOS MSE	0.432	-0.93%	-0.69%	0.46%	-3.94%	-1.62%	-4.86%	-1.62%	-1.85%	-7.87%	3 Lags	PMI, New Orders, Inventories
DM Test P-Value		0.106	0.469	0.179	0.084	0.001	0.056	0.651	0.001	0.005		

Notes: This table reports out-of-sample mean squared errors of regressions of month-to-month percentage change of industrial production on a set of real-time predictors of IP from 2018m1-2020m1. The text measures represent the survey response sentiment using either dictionary-based methods (columns 2-5), transfer learning of financial BERT models (columns 6-7), or fine-tuned models trained on a random selection of human-labeled ISM survey responses (columns 8-9) or naturally occurring labels predicting future production (column 10). PMI, New Orders, and Inventories are monthly diffusion indexes released by the ISM at the beginning of the month. The 3 lags are the initial estimate of IP_Growth for month t-1, the revised estimate of IP_Growth for month t-2, and the twice-revised estimate of IP_Growth for month t-3. The P-values are calculated using the Diebold-Mariano out-of-sample error statistics. Significance levels are indicated by *** (1 percent), ** (5 percent), and * (10 percent).

Table 7: Out-of-sample Regression Results (Global Financial Crisis)

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)		
In-Sample: 2001M11-2007M11 Out-of-Sample: 2007M12-2009M6												
		Dictionary Based Methods				*Deep Learning Methods*					Controls Included	
Text Measure	*Baseline*	LM	Harvard	AFINN	Stability	FinBERT (v1)	FinBERT (v2)	TF-Small	Fine-Tuned BERT: Human Labeled Data	Fine-Tuned BERT: Production Data	IP Lags	ISM Variables
OOS MSE	1.933	-17.23%	-8.85%	-6.78%	-9.98%	-5.12%	-8.64%	-17.38%	-22.19%	-1.50%	3 Lags	PMI
DM Test P-Value		0.117	0.079	0.128	0.116	0.068	0.141	0.048	0.045	0.155		
OOS MSE	1.584	-13.19%	-3.85%	-4.61%	-7.64%	-4.86%	-5.81%	-13.51%	-16.67%	-1.01%	-	PMI
DM Test P-Value		0.192	0.224	0.252	0.158	0.217	0.408	0.113	0.091	0.199		
OOS MSE	2.225	-20.63%	-6.70%	-8.67%	-9.53%	-16.22%	-18.74%	-0.76%	-8.09%	-1.35%	3 Lags	-
DM Test P-Value		0.094	0.092	0.104	0.102	0.000	0.022	0.795	0.16	0.197		
OOS MSE	1.622	-11.59%	-4.81%	-2.03%	-0.12%	0.12%	-2.65%	-13.69%	-11.47%	-10.11%	3 Lags	New Orders
DM Test P-Value		0.127	0.098	0.134	0.802	0.948	0.275	0.060	0.101	0.043		
OOS MSE	1.669	-11.26%	-4.97%	-1.80%	0.66%	0.12%	-2.34%	-16.66%	-14.92%	-11.44%	3 Lags	PMI, New Orders, Inventories
DM Test P-Value		0.178	0.12	0.284	0.602	0.92	0.348	0.035	0.056	0.059		

Notes: This table reports out-of-sample mean squared errors of regressions of month-to-month percentage change of industrial production on a set of real-time predictors of IP from 2007m12-2009m6. The text measures represent the survey response sentiment using either dictionary-based methods (columns 2-5), transfer learning of financial BERT models (columns 6-7), or fine-tuned models trained on a random selection of human-labeled ISM survey responses (columns 8-9) or naturally occurring labels predicting future production (column 10). PMI, New Orders, and Inventories are monthly diffusion indexes released by the ISM at the beginning of the month. The 3 lags are the initial estimate of IP_Growth for month t-1, the revised estimate of IP_Growth for month t-2, and the twice-revised estimate of IP_Growth for month t-3. The P-values are calculated using the Diebold-Mariano out-of-sample error statistics. Significance levels are indicated by *** (1 percent), ** (5 percent), and * (10 percent).

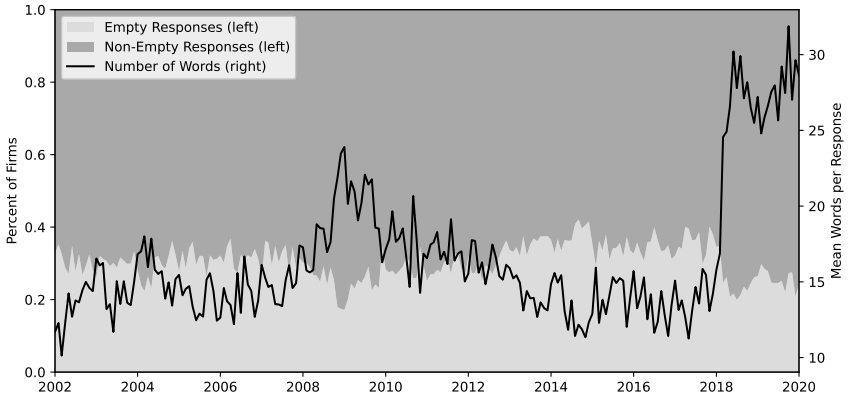
Table 8: Average Net Positive Scores

Positive Words	Score	Negative Words	Score
specials	0.055	weak	-0.063
improved	0.053	inability	-0.064
excellent	0.051	fragile	-0.064
booming	0.049	decline	-0.066
upbeat	0.048	downward	-0.066
improves	0.048	declining	-0.068
improvement	0.047	downs	-0.069
improve	0.046	weakening	-0.070
increase	0.045	depressed	-0.071
good	0.044	weaken	-0.072
rum	0.043	discontinued	-0.073
launch	0.041	slow	-0.075
brisk	0.040	offs	-0.075
increased	0.040	insufficient	-0.076
increasing	0.036	instability	-0.080
heightened	0.033	slowing	-0.081
upgrade	0.033	slug	-0.084
advantages	0.033	erosion	-0.085
lift	0.032	errors	-0.093
doubled	0.032	unstable	-0.105

Notes: Words are those with the most positive and most negative scores, among words appearing more than 5 times in the data. The "score" is the net positive probability from the Shapley decomposition: The average marginal contribution of the word toward a positive classification, minus the average marginal contribution towards a negative classification.

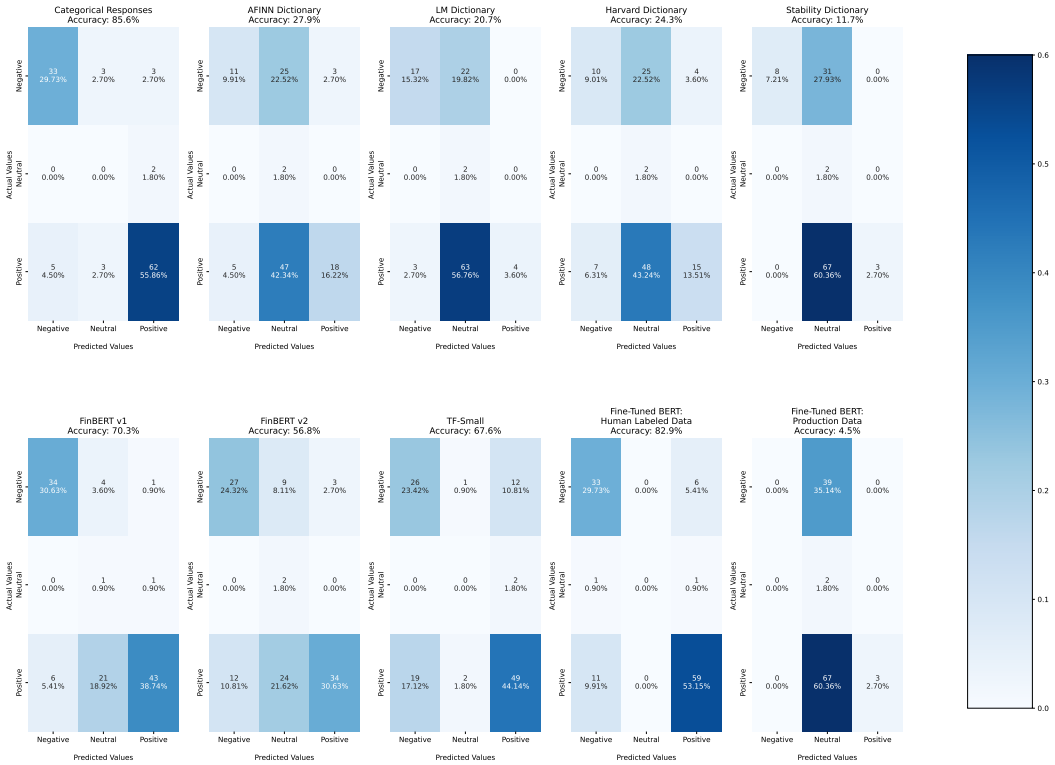
Figures

Figure 1: ISM Survey Responses



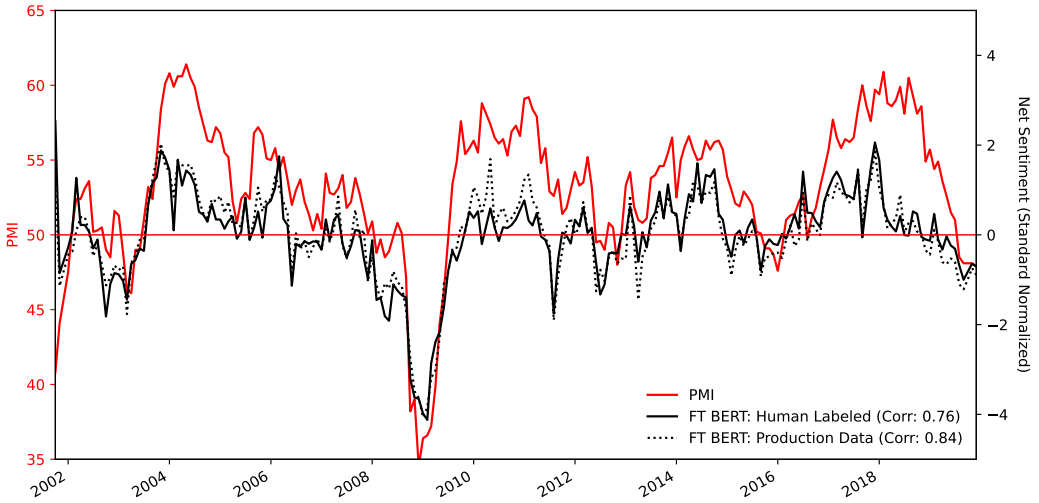
Notes: This figure shows the percentage of firms and the word counts for the ISM survey responses. [Left Axis] The light (dark) grey region shows the percent of firms that provided empty (non-empty) responses on their monthly response. [Right Axis] The black line shows the mean number of words per response across all respondents for a given month.

Figure 2: Confusion Matrices and Accuracy Scores



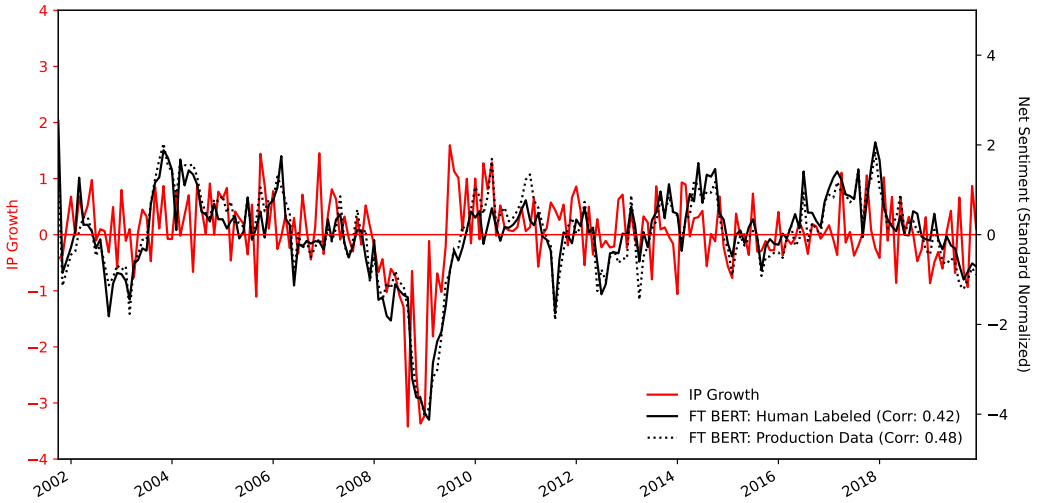
Notes: This figure shows the confusion matrix for nine manufacturing sentiment measures applied to the training dataset of manually labeled ISM survey responses. The rows of each matrix refer to the actual values, while the columns refer to the predicted values. Values along the diagonal are correctly classified, while values on the off-diagonals are incorrect. The shaded color refers to the percentage of responses within a given cell, according to the heatmap legend on the right.

Figure 3: ISM PMI and Sentiment Indexes



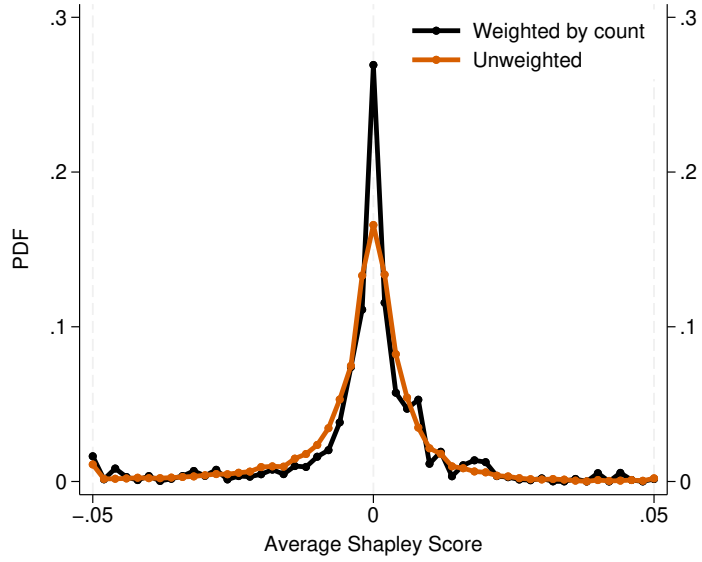
Notes: This figure shows two manufacturing sentiment measures alongside the ISM PMI (red). Fine-Tuned BERT: Human Labeled Data and Fine-Tuned BERT: Production Data are fine-tuned BERT models trained on human-labeled sentiment and future firm-level production data, respectively. Correlations between the two sentiment measures and the ISM PMI are provided in parentheses.

Figure 4: Industrial Production and Sentiment Indexes



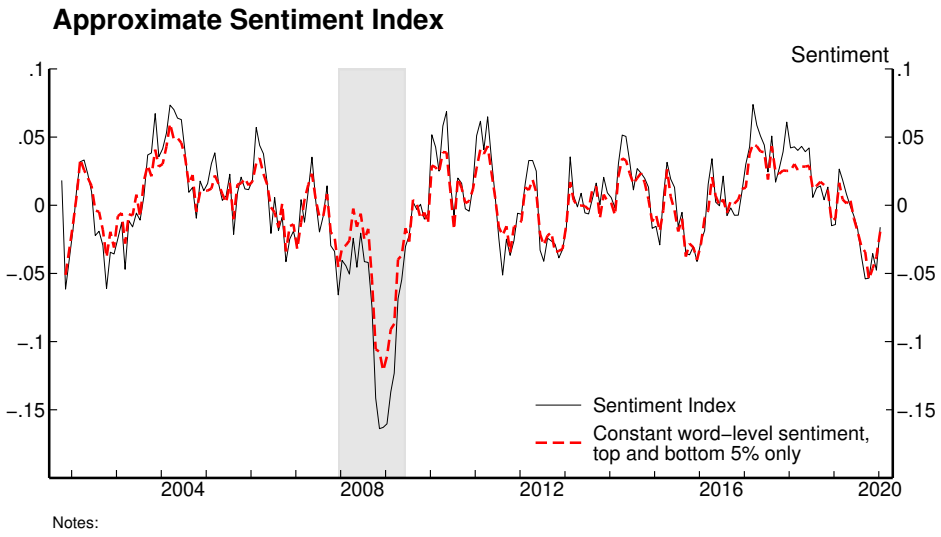
Notes: This figure shows two manufacturing sentiment measures alongside IP Growth (red). Fine-Tuned BERT: Human Labeled Data and Fine-Tuned BERT: Production Data are fine-tuned BERT models trained on human-labeled sentiment and future firm-level production data, respectively. Correlations to IP Growth are provided in parentheses.

Figure 5: Token PDFs



Note: Distribution of tokens across Shapley scores. For this graph, Shapley scores are Winsorized at the top and bottom 5 percent of the distribution. "Unweighted" gives the distribution of unique tokens by score, "Weighted" gives the distribution weighted by number of appearances in the data.

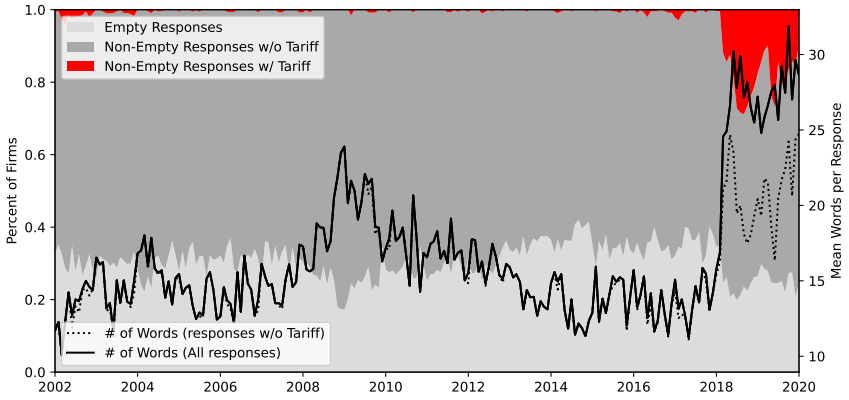
Figure 6: Approximate Sentiment Index



Note: Sentiment index (black line) is based on BERT, as fine-tuned used future firm-level production data. The approximate index (red dashed line) is calculated by (1) estimating Shapley decompositions to obtain each word's contribution to the score of each comment, (2) averaging those scores over comments to get time-invariant word scores, and (3) only keeping the top and bottom 5 percent of words.

Appendix

Figure A1: What Explains the 2018 Increase in Text Responses?



Notes: This figure shows the percentage of firms and the word counts for the ISM survey responses. [Left Axis] The light (dark and red) grey region shows the percent of firms that provided empty (non-empty) responses on their monthly response. The red region highlights the number of firms that included the word "tariff" in their response. [Right Axis] The solid (dotted) black line shows the mean number of words per response across all respondents (excluding responses using the word "tariff") for a given month.

1 Methods

1.1 Dictionary Based Methods

A bag of words dictionary method is a mapping of the form $f : \mathbb{R}^V \to \mathbb{R}$ where $x^d \in \mathbb{R}^V$ is a V-dimensional vector and V represents the size of the set of the unique tokens across a corpus, S. The elements of x^d, e.g. $x^d_{w_i}$, represent the number of occurrences of the word w_i in document d.

To implement a dictionary method, we select a subset of the unique words across the corpus, $D \subset S$. Then, the function f is simply the sum of the elements in x^d, i.e. $f(x^d) = \sum_{w_i \in D} t_{w_i} x_{w_i}$, where t_{w_i} represents the weight given to word w_i. Typically, for sentiment analysis, there are three weights: +1 for positive words, 0 for neutral words, and -1 for negative words inside of D.

1.2 BERT Models

This subsection describes the basics of BERT, one of the most popular transformer-based models. It is difficult to explain transformer-based models briefly, in part because they are fundamentally complex. Existing descriptions of these models are either very terse, assume extensive knowledge of deep learning terminology and history, or are vague. Our goal is to provide a reasonably succinct overview of the architecture, accessible to someone not specialized in deep learning.

These models are called "transformers" because the input is transformed into a representation in a latent space.[18] This aspect of the architecture is not particularly unique; the main distinguishing feature of transformers is *attention*, a mechanism that allows the interpretation of words in a sentence to be influenced by the other words in the sentence.

Transformers gained popularity in part because they showed excellent performance on a wide variety of language tasks and, relatedly, the design allows for extreme parallelism.

Section 1.2.1 describes in detail the mechanics of what happens when text is fed to a BERT model used for sentiment classification (or more broadly, any type of classification). Section 1.2.2 goes over how BERT models are trained. Section 1.2.3 discusses how the

[18] In the original transformer paper the application was machine translation. The input, in one language, was transformed ("endcoded") into an abstract representation and this was then "decoded" back into the second natural language. The BERT architecture only includes the encoding step, and classification or other tasks use the abstract representation as an input. GPT-like models are considered decoder-only models, which seek to generate the next word in a sequence using a representation of the sequence so far.

BERT model is further trained and specialized ("fine-tuned") to perform specific tasks or use additional data.

1.2.1 BERT at Inference

BERT at inference can generally be defined in five steps. First, the input text is partitioned into its atomic unit, e.g. a word, in a process known as tokenization. Each token is represented in an abstract vector space that captures the syntactic and semantic meaning of the token. Second, the word order is taken into account using positional embeddings. Third, the model adjusts the attention it should place to other words in the sequence through the defining characteristic of transformers, known as the *attention mechanism*. Fourth, a normalization step concatenates the attention with the input embeddings. Lastly, the new representation of the input sequence is used for sentiment classification. We guide the reader through these five steps below.

Step 1: Creating the Input Embeddings

A transformer-based sentiment model can be defined as a mapping of a fixed number of tokens, L, such that $f_T : \mathbb{R}^{V \text{x} L} \rightarrow \mathbb{R}$, where the input x is a $V \times L$ matrix.

A *token* is a word, a part of a word or a single character. V is the size of the *vocabulary*, the tokens that are valid inputs.[19] The columns of x are dummy vectors of size V, with element i equal to 1 if the word in the $i-th$ position is equal to w_i, and zero otherwise. Many pre-trained BERT models fix L, the sequence length, at 512 tokens. If a sequence contains less than 512 tokens, then the remaining sequences are "padded", in other words replaced with a special "end of sequence" token that will mask any parameters associated with those positions. If a sequence has more than 512 tokens, only the first 512 would be used.

Transformers, like most NLP methods, represent words as vectors, called embeddings. In large-scale, general versions of BERT, such as the base version released by Meta,[20] the word (token) is represented as a 768-dimensional vector. The high dimensionality should help capture the fact that words' meanings have many dimensions, so two words can be similar in many ways but still distinct along important dimensions.

[19]BERT has a vocabulary of 30,522 tokens. These tokens include most common words, and "token" is sometimes used interchangeably with "word". But, importantly, the vocabulary also includes many word parts, such as common word endings, and all single characters. Thus BERT can process any text, since unfamiliar words can be built up from word fragments and single characters.

[20]`https://github.com/google-research/bert/blob/master/README.md`

At inference time the embeddings are fixed. The first step of f_T is to convert the VxL input into a NxL matrix, where each token indicator column (of length V, the size of the vocabulary) is converted into a length N word embedding vector. Define the NxL matrix as x'.

Step 2: Adjust to Generate Positional Embeddings

Transformer models do not inherently account for the order of the inputs anywhere in their architecture, a characteristic that is critical for understanding the meaning of text. Adding an index number of the input token (e.g. 1 for the first token, 2 for the second token, etc.) would create two difficulties. First, this method leads to unbounded positional adjustments. Second, the model may not be able to generalize for sequence lengths that are rarely seen, especially longer sequences. The model could see plenty of first word adjustments, second word adjustments, etc. but larger values would become rarer. The typical solution to account for positions is to use sine and cosine functions. For input token x'_k, an N-dimension vector - p^k- is generated. For $0 \leq i < N/2$, :

$$\begin{aligned} p^k_{2i} &= sin(\frac{k}{10000^{\frac{2i}{N}}}) \\ p^k_{2i+1} &= cos(\frac{k}{10000^{\frac{2i}{N}}}) \end{aligned} \tag{3}$$

where p^k_i is the i-th index of p^k and N is the dimension size of the target embedding.

We adjust the column vectors of x' for their position by adding p to x'. Call the adjusted matrix $y \in \mathbb{R}^{N\text{x}L}$

Step 3: Attention Mechanism

Next we enter the transformer block. This is a mapping $f : \mathbb{R}^{N\text{x}L} \to \mathbb{R}^{N\text{x}L}$. Note that the output and input are the exact same size. This step of the transformer model is arguably the most important as the final representation of the word vectors captures well the meaning of the text.

We begin by creating a set of key, value, and query matrices. This step mimics a look-up table in a database table. They are defined as follows:

$$
\begin{aligned}
K(y_i) &= W_k y_i \\
V(y_i) &= W_v y_i \\
Q(y_i) &= W_q y_i
\end{aligned} \tag{4}
$$

where $y_i \in \mathbb{R}^N$ (a column vector from the input y) and $W_k, W_v, W_q \in \mathbb{R}^{M\text{x}N}$. Ultimately, the resulting vectors $K(y_i), V(y_i), Q(y_i) \in \mathbb{R}^M$ are transformations of the input vector, y_i. This can be thought of as projecting the individual vector y_i into into an abstract M-dimensional space. Using the entire L-length input sequence, an LxL matrix is then created:

$$
\alpha = \begin{bmatrix} \alpha_{11} & \dots & \alpha_{1L} \\ \vdots & \ddots & \vdots \\ \alpha_{L1} & \dots & \alpha_{LL} \end{bmatrix} \tag{5}
$$

where $\alpha_{i,j} = \text{softmax}_j(\frac{Q(y_i) \cdot K(y_j)}{\sqrt{M}})$ (i.e. the rows of the α matrix sum to 1). Essentially, $\alpha_{i,j}$ measures how similar the query is (a transformation of the i-th word in consideration) to the other keys (a transformation of the other words in the sequence).

Each vector is then weighted depending on the attention of that word with the other words in the sequence.

$$
u_i' = W_0 \sum_{j=1}^{L} \alpha_{i,j} V(y_j) \tag{6}
$$

where $W_0 \in \mathbb{R}^{N\text{x}M}$ and $u_i' \in \mathbb{R}^N$. This assumes there is just one head. However, we can have multiple heads such that equations (4), (5), and (6) are repeated with H different sets of parameters. For example, for head h, the W matrices in (4) will be different: $W_{k,h}, W_{v,h},$ and $W_{q,h}$. This will lead to $\alpha_{\mathbf{h}}$ in (5), and (6) will become:

$$
u_i' = \sum_{h=1}^{H} W_{0,h} \sum_{j=1}^{L} \alpha_{i,j}^{(h)} V^{(h)}(y_i) \tag{7}
$$

with $W_{0,h} \in \mathbb{R}^{N\text{x}M}$.

The last step of the attention mechanism is to add back the resulting matrix u_i' from (7)

back to the input vector y_i, and then pass the resulting vector through a layer normalization function, which is analogous to a standard normalization procedure but slightly adjusted with a different scaling and shifting parameter.[21]

$$u_i = \text{LayerNorm}(y_i + u_i') \tag{8}$$

Step 4: Feed Forward and Normalize

Next the resulting vector, u_i, is passed to a ReLU network, then added to itself, and finally normalized once more:

$$z_i' = W_2\text{ReLU}(W_1 u_i) \tag{9}$$

$$z_i = \text{LayerNorm}(u_i + z_i') \tag{10}$$

where $W_1 \in \mathbb{R}^{P\text{x}N}$ and $W_2 \in \mathbb{R}^{N\text{x}P}$. The final vector $z_i \in \mathbb{R}^N$ is the transformed input vector y_i that accounts for the position of the i-th word and the attention the word emits and receives from other words in the sequence.

Step 5: Sentiment Classification

The last step entails a mapping $f : \mathbb{R}^{N\text{x}L} \to \mathbb{R}$. Typically, this is a neural network that takes as input a matrix and outputs a probability distribution across 3 categories: positive, negative, and neutral.

1.2.2 Training

As with most deep learning models, BERT is estimated using stochastic gradient descent. Model weights are adjusted using a learning rate, λ, such that $w_{i+1} = w_i - \lambda\frac{\delta L}{\delta w_i}$, where L is the loss function. If λ is too large, updates may exceed w_i and the optima may be missed. Setting λ too small may lead to smaller adjustments and more time needed for convergence. To accelerate the process and improve the efficiency of finding optimum weights, an extension of gradient descent, known as Adaptive Moment Estimation (or the ADAM optimizer), is typically used.

[21] The layer normalization function has two hyperparameters, γ and β, and is defined as follows: $\text{LayerNorm}(x;\gamma,\beta) = \gamma * \frac{x-\mu}{\sigma} + \beta$

For financial text sentiment classification, two popular BERT models have been pre-trained on large corpi of data and are publically available: Huang et al. (2022) (which we refer to as FinBERTv1) and Araci (2019) (which we refer to as FinBERTv2). FinBERTv1 was trained on nearly 10,000 sentences from SEC filings, equity reports, and earnings conference call transcripts that were hand labeled for sentiment. FinBERTv2 was trained on nearly 5,000 randomly selected sentences from financial news articles, and nearly 1,000 financial news tweets, all of which were manually labeled for sentiment.

1.2.3 Fine-Tuning

We fine-tune BERT in three different ways. The first two use human-labeled responses from a sample of the ISM survey, while the last exploits the panel structure of the survey to fine-tune a model with naturally occurring data.

TF-Small and Fine-Tuned BERT: Human Labeled Data

We first create a dataset of responses that were hand-labeled for sentiment. We format the ISM survey responses at the firm-month-question level and randomly select 1,000 text responses. Each response was classified for sentiment by two economists using the following question as a guide: "Is this comment consistent with manufacturing IP rising month over month?" The classifications were either positive, neutral, or negative. We keep only 700 responses for which both economists agreed on the sentiment.

Then, we split our sample such that 90 percent is used for fine-tuning, and 10 percent is leftover for an unseen test set for sentiment model comparisons (i.e. is never used for the training). We use this human-rated training data to train two types of models. For the first model, we train a plain vanilla transformer model from scratch using a simple architecture (with only one head and embedding dimensions of size 12-16). We call this model simply the *TF-Small* model (TF for "transformer"). The second model uses the pre-trained, off-the-shelf BERT as the baseline transformer, but we fine-tune the last layer using our human-labeled dataset. We call this model *Fine-Tuned BERT: Human Labeled Data*. Note that for the *TF-Small* model, we estimate the entire attention mechanism weights, whereas for *Fine-Tuned BERT*, we are further tuning the attention weights that were pre-trained on a large dataset.

Fine-Tuned BERT: Production Data (Naturally Occurring Data)

The third fine-tuned model capitalizes on the panel structure of the ISM survey. Since same firms appear in the survey over time, we use pre-trained BERT to estimate a model that predicts the production categorical response ($PROD_{f,t+1}$) from the previous months text ($text_{f,t}$). The availability of the target variable, $PROD_{f,t+1}$, is beneficial for us in two ways. First, rather than manually labelling hundreds or thousands of responses, we obtain **naturally occuring data** that is several orders of magnitude larger than our human-labeled dataset, at nearly zero-effort and instantaneous availability. Second, the task of predicting the production categorical response one month ahead perfectly aligns with our downstream task of forecasting aggregate industrial production.