News sentiment

Benjamin Golez, Rasa Karapandza**, & Frederik Wisser†*

June 2023

Abstract

We introduce a novel method for training computer algorithms to measure news sentiment. Our approach leverages human-coded sentiment scores from over 200,000 newspaper articles to teach the computer to select words, word combinations, and their linear weights. In an out-of-sample test, examining newspaper articles about US companies, we show that: (i) our news sentiment metric displays a bimodal distribution similar to that observed in the human-coded sentiment scores, (ii) our news metric outperforms the widely-used bag-of-words approach and recent machine learning models in explaining human-coded news sentiment, and (iii) our news sentiment metric serves as a robust predictor for daily stock returns.

Keywords: Textual analysis, Machine learning, News sentiment, Stock returns

JEL Classification: C53, C55, G11, G12, G14, G17, G41

* Mendoza College of Business, University of Notre Dame, Notre Dame, IN 46556, USA;
Tel. +1-574-631-1458, <bgolez@nd.edu>
** Department of Finance, Accounting & Real Estate, EBS Business School, Gustav-Stresemann-Ring 3, 65189 Wiesbaden, Germany; Tel. +49-611-7102-1244, <rasa.karapandza@ebs.edu>; and
Division of Social Science, New York University Abu Dhabi, <karapandza@nyu.edu>
† Department of Finance, Accounting & Real Estate, EBS Business School, Gustav-Stresemann-Ring 3, 65189 Wiesbaden, Germany; Tel. +49-611-7102-1244, <wisser.frederik@ebs.edu>

I. Introduction

The use of text as data is becoming increasingly popular, and news sentiment is often used in finance to gauge investor sentiment (Gentzkow, Kelly, Taddy, 2019). However, quantifying news sentiment is complex because the text is high-dimensional. Not surprisingly, researchers often find a weak association between the measured news sentiment and stock returns (Tetlock, 2008).

The traditional method to measure news sentiment in finance is the "bag of words" approach. Words are assigned either a positive, a negative, or a neutral tone, and the sentiment is expressed as a ratio of positive and negative words (Tetlock 2008; Loughran and McDonald 2011). The appeal of this approach lies in its simplicity. It is easy to understand, reproduce, and amend the dictionary to any application. However, the simplicity of this approach comes at the cost of noise in choosing the words that determine the sentiment and the difficulty in determining the meaning of more structured sentences.

An alternative approach is to adopt machine learning and let the computer pick the words that determine the news sentiment (Gentzkow, Kelly, Taddy, 2019). This approach is very flexible as it allows for word combinations and larger weights for essential words. As such, machine learning is very appealing and has attracted much interest. The recent rise of Large Language Models like Chat GPT has made machine learning even more popular (Lo and Singh, 2023). However, the implementation of machine learning is not straightforward.

The main challenge is determining the objective function for the computer to assign the sentiment. Ideally, the computer should choose the words to maximize the fit to the "true" sentiment of news. For lack of such a measure, researchers often let the computer choose the

words that maximize some other objective, e.g., words that maximize stock return predictability (Ke, Kelly, Xiu 2019; Garcia, Hu, Rohrer 2023). This is a sensible choice if the objective is to find words associated with stock price movements. However, there is no guarantee that this procedure captures the "true" sentiment. If we want to test whether news sentiment predicts stock returns, we need to train the computer on an objective measure of news sentiment that is not endogenous to the very object we want to predict.

In this paper, we make a step in this direction. We use proprietary data on human-coded sentiment scores from newspaper titles as a training sample for machine learning. Our training set is large (it covers over 200,000 newspaper titles), and it is based exclusively on newspaper articles about companies. Importantly, individuals whose full-time job is to read and assess the message of newspaper articles determine the sentiment score. All coders undergo rigorous training to guarantee the consistency of coding procedures. Their job of determining news sentiment has no direct link to the stock market. We thus have a measure of news sentiment, which is arguably the closest to the objective measure of news sentiment.

We start our analysis by discussing the distributional characteristics of our human-coded news sentiment. Our data cover 213,186 newspaper articles containing news about more than 200 companies. Newspaper articles are published in English-speaking countries (US, UK, Canada, and Australia) between 2007 and 2016. The news sentiment score is coded on a scale from – 4 (most negative) to + 4 (most positive).

We first analyze the distributional characteristics of human-coded news sentiment. The observation that stands out the most is that human-perceived news sentiment has a bi-modal distribution. In a frequency histogram, we observe two spikes, one at a news sentiment score

of – 2 and another at + 2. That is, most news articles either have a pronounced positive or pronounced negative sentiment. This is consistent with the notion of "sensationalism" and confirms that newspapers tend to publish news that has a strong sentiment charge. However, we note that the most common article news sentiment score is + 2, which is contrary to the popular belief that newspapers publish primarily negative news. We assert that a bimodal distribution of news sentiment is a vital data characteristic that news sentiment methods should strive to replicate.

Next, we use newspaper article titles (news headlines) from our human-annotated data to train the computer to pick words and word combinations, along with their weights. We focus on the titles of the articles rather than full articles since titles are shorter and coded separately from the main article. The fact that titles are shorter makes it easier for the machine to associate the news sentiment score with specific words and word combinations. Titles are also carefully crafted by journalists and editors to reflect the overall sentiment of news articles. We estimate the sentiment charge of each word and word combination in a logit model by regressing the human-coded sentiment score on the counts of our filtered single words and word combinations. The list of words the computer designates as positive and negative is intuitive. The most positive single word is "safest," and the most negative is "scandal." For word combinations, most positive and negative words are "found not (guilty)" and" biggest loss," respectively.

Since we train the computer on the article titles, we then use the main body of news articles as an out-of-sample test. We find that our newly constructed news sentiment explains

42% of the variation in the human-coded news sentiment. Our method also produces a bi-modal distribution of news article sentiment, similar to the one observed in the human-coded data.

Finally, we test whether our news sentiment predicts daily stock returns. We run a regression of open-to-close daily stock returns on our company-level daily news sentiment. We control for all the standard variables and cluster standard errors by trading day and by the company (like Tetlock 2008). We find that our news sentiment metric is positively and significantly related to future daily stock returns (*t*-statistic of 2.42). We also create a long-short portfolio, where we buy companies with the most positive news sentiment and short companies with the most negative news sentiment. We rebalance portfolios daily. The long-short portfolio has an annualized alpha of 12.5% (*t*-statistic of 2.45).

Overall, we find that our newly constructed measure of news sentiment captures bimodality in human-coded news sentiment and that it predicts stock returns in the cross-section. Next, we run a horse race with the existing methods for determining news sentiment.

We start by comparing our results to the traditional "bag-of-words" method based on the single words list of Loughran and McDonald (2011). We use either percentage of negative words or the difference between the negative and positive words. These methods explain 35% and 37% of the variation in the human-coded news sentiment. However, neither of the two alternatives produces a bimodality of news sentiment. Compared to our method, they are also weaker predictors of daily stock returns (*t*-statistics of -1.42 and -1.67).

Next, we compare our results to the sentiment scoring algorithm developed by OpenAI (the company behind ChatGPT). The OpenAI states the algorithm is trained on Amazon

reviews and achieves state-of-the-art sentiment analysis accuracy.[1] When applied to our sample of news, it explains 41% of the variation in human-coded news sentiment. However, it does not create bimodality in the distribution of sentiment scores, and it fails to predict stock returns (t-statistic of -0.74).

Finally, we apply the FinBERT, a pre-trained natural language processing model of Araci (2019). It is built by fine-tuning the BERT model for financial sentiment classification. However, when applied to our sample of news, we find that the distribution of news sentiment is unimodal, and the variation in the news does not predict daily returns (t-statistic of 1.36).

The fact that none of the alternative methods produce bimodality in news sentiment scores (or reliably predict stock returns) may be surprising initially. The bag-of-words approach we implement is very popular in the literature, and the NLP algorithms we consider are state-of-the-art. However, more and more studies emphasize that sentiment analysis is domain-specific and that general sentiment analysis, no matter how advanced, may result in arbitrarily low power when applied to a specific domain. That is why Loughran and McDonald (2011) adjusted the general Harvard list of positive and negative words to finance applications. Using similar arguments, Araci (2019) fine-tuned the BERT model to adjust it to capture sentiment in financial statements. The language in newspapers is even more specific. When looking at our list of most positive and negative words, we notice words like “sweeps”, “crowned”, “heaven”, “cheated”, “deception”, which suggests that our word list loads firmly on a

[1] Https://openai.com/research/unsupervised-sentiment-neuron.

journalistic jargon. The presence of newspaper jargon is even more noticeable when we look at word combinations and notice expressions such as “hot seat”, “come clean,” and takes heat”. None of the alternative methods is fine-tuned for news. The sentiment scoring algorithm of OpenAI is trained on Amazon reviews. The LM word list is designed for 10K statements, and FinBERT is mainly trained on formal financial text. Thus, our study reinforces the notion that sentiment analysis is domain specific. If we want to analyze news sentiment, we must train the computer in newspaper language.

We also find that it is essential to allow for word combinations. Our method differs from the conventional bag-of-words approach in two main ways: (i) we allow the computer to choose words and word combinations, and (ii) we let the computer determine the weights of each word or word combination. To assess which difference is more important, we repeat the analysis by restricting the computer to single words from the LM dictionary, but we let the computer assign the weights to each word. We find that such news sentiment does produce bi-modality in news sentiment scores, but the distribution is highly right-heavy. Moreover, such news sentiment performs worse in explaining the human-coded sentiment scores, and it fails to predict stock returns. This confirms that word combinations have several advantages over single words (see also a recent paper by Garcia, Hu, Rohrer, 2021). They allow us to account for the effect of negations. For example, we find that negations of positive words, such as “not supported”, “not relief,” and “not worth,” are all in the top decile of features associated with negative human-coded news sentiment. Word combinations also allow us to capture meaning from structures that are incorrectly classified when using single-word dictionaries. For instance, the sentence “the case against the company was dropped” would be erroneously

assessed as highly negative since "dropped" and "against" are negative words according to the LM dictionary. Yet when we allow word combinations, we find that "dropped against" is among the top 15 features associated with positive human-coded news sentiment. Finally, we find that words that have a neutral meaning (and are excluded from traditional dictionaries) can carry a strong sentiment charge when combined with other words (e.g., "record sales", "ends probe", "record fine", and "pays price").

We contribute to the literature analyzing the relationship between text-based investor sentiment and stock prices (Tetlock 2007; Tetlock, Saar-Tsechansky, and Macskassy 2008; Loughran and McDonald 2011; Loughran and McDonald 2016; Gentzkow, Kelly, Taddy, 2019; Ke, Kelly, Xiu 2019; Garcia, Hu, Rohrer 2021). Our main innovation is to train the computer algorithms on a large scale of human-annotated news sentiment scores to create a new list of words and word combinations (and associated weights) for determining news sentiment.[2] We show that our news sentiment measure captures bi-modality in human-coded news sentiment scores. Applying our approach to a broad scope of newspaper news about S&P 500 companies, we find that news sentiment reliably predicts daily returns in the cross-section of stocks. News sentiment is, therefore, important for stock price movements.

[2] We will make publicly available all the single words and word combinations along with their weights. Implementing out sentiment measure is as simple as a traditional bag-of-word approach, except that our list of words is different and each word has a specific weight determining its importance (the traditional bag-of-words approach assumes equal importance for all listed words).

Our results emphasize the importance of training the machine on a domain-specific sample. Since journalist language is rather specific, training the computer on a news-based training sample is important. This resonates with the argument from computer science literature that fine-tuning models for domain-specific applications yields greater improvements than increasing the complexity of machine learning models (Araci 2019).

The rest of the paper is organized as follows. In Section II, we describe the data sources and discuss the summary statistics. In Section III, we present the methodology. In Section IV, we present the main results. Section V concludes.

II. Data and Descriptive Statistics

This section describes the primary data on which we train our model. We also describe newspaper data on which we test the models. Finally, we list any additional data sources.

A. Prime Research Data

We train our model using human-annotated new sentiment score data. This data set comes from Prime Research (PR). PR is a leading global media monitoring and analysis provider for international companies and institutions. The company has been in business since 1987 and employs over 1,000 data analysts in eight research centers in Europe, North America, and Asia. It constantly monitors news reports about its clients across countries and languages in media outlets. PR started with analyzing conventional media outlets for the automotive industry, which remains its main specialization. The PR automotive data has been previously used by Golez and Karapandza (2022). In this paper, we use PR data for the automotive industry and all other available industries,

as long as the news is reported in an English-speaking country. In total, we have 213,186 unique articles about over 300 companies mentioned in the major news outlets in Australia, Canada, the United Kingdom, and the United States from 2007 to 2016.

PR's standard approach is for native speakers to code all news manually. Coders split each news article into several segments (e.g., a title, a paragraph, or a part of a paragraph) and determine the news sentiment score for each segment that contains a self-contained message. The news sentiment score is assigned to all segments that contain value judgments and is evaluated on a 9-point scale that ranges from −4 (most negative) to +4 (most positive); the middle value (0) represents a neutral sentiment score.

The titles of articles (headlines) are of special interest to our study. Titles are treated as separate segments and come with separately annotated sentiment score that is based solely on the message contained in the title of the article. Since titles are typically short (from a few words to a maximum of a sentence), they are well-suited for our purpose: a small number of words per score reduces the noise-to-signal ratio when associating words with sentiment.

Panel A of Table 1 reports summary statistics for all the PR newspaper articles in our sample. Panel B reports the same statistics for the article titles only. The number of companies covered varies from 168 at the beginning of our sample (in 2007) to 278 at the end of our sample period (in 2016). The number of articles per year varies between 9,429 and 42,397. Around 70% of the articles are about the automotive industry. Among the rest of the industries, the financial and pharmaceutical industries have the best representation. For newspaper articles, the average sentiment score is between 0.31 and 0.93. The summary statistics for

newspaper article titles are very similar, with the average sentiment score between 0.33 and 1.01. The median sentiment score is somewhat higher for article titles than the articles.

[Insert Table 1 and Figure 1 here]

In Figure 1, we plot the distribution of news sentiment scores separately for newspaper articles (Panel A) and article titles (Panel B). There are two modes in the distribution of the sentiment scores—the number of observations spikes at news sentiment score -2 and news sentiment score +2. The number of observations with neutral sentiment scores is low. This is consistent with news sensationalism, an editorial tactic to select stories that excite the largest number of readers. While an incentive to report sensational news is well-understood, we are unaware of any other paper documenting that news sentiment scores follow a bi-model distribution.[3] Still, we do not see many extreme observations. The number of observations with sentiment scores of -4 or +4 is relatively low. The news sentiment also seems skewed toward being more positive (the mode is a +2), which is surprising in light of the public perception that newspapers mainly focus on negative reporting.

B. Factiva Data

To test the models, we obtain newspaper articles from Factiva. Specifically, for every S&P 500 company as of January 2007, we download all newspaper articles published by the WSJ, the *New York Times*, *USA Today*, or the *Washington Post* from 2007 through December

[3] For sensationalism in news about mergers and acquisitions, see Ahern and Sosyura (2015).

2017. In total, we obtain 69,157 articles. A given article can appear more than once if it mentions two (or more) companies in the S&P 500 universe. We retain articles with multiple firm tags and match each tagged company to the corresponding article. In total, we have 200,642 article-company pairs. We make several checks in the process of cleaning the data to make sure that we keep only relevant articles.

Table 2 reports summary statistics of our final Factiva data set. The number of articles varies between 7,668 and 5,274 per year. The number of companies covered varies between 376 and 265 per year.

[Insert Table 2 here]

C. Other Data

We obtain the S&P 500 Index constituent firms and their associated stock returns and stock market trading volume from the CRSP database. We obtain fundamental accounting information from the Compustat database. The Fama–French factors and the risk-free rate are downloaded from Kenneth French's website.[4] To count the number of negative and positive words, we use the finance word dictionary of Loughran and McDonald (henceforth LM), which we download from their website.[5]

[4] https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

[5] The dictionaries can be found on the following website: https://sraf.nd.edu/textual-analysis/resources/.

D. Text Pre-processing

We employ standard text-cleaning procedures for the news data: (i) we remove all URLs, (ii) we remove all numbers, (iii) we change all words to lowercase, (iv) we replace multiple white spaces with a single space, (v) we expand contractions (e.g., replace "hasn't" with "has not"), and (vi) we remove possessives ("'s") and hyphens. We also remove common English stop-words.[6] Since standard stop-word lists contain negations, we remove stop-words only if they are *not* a negation. Negations play a vital role in the analysis because our approach allows for directly interpreting negated words.

We also implement an additional text pre-processing step. Since we allow for word combinations (on top of single words), we standardize negations to "not." That is, we replace the many different forms of negations, such as "neither", "never", "no", "nobody", "none", and "no one" with "not". This ensures that our model does not distinguish between word combinations like "neither did they succeed" and "they did not succeed", thereby substantially reducing the dimension of the final word space.

III. Methodology

We start this section by describing our primary model for measuring news sentiment, which we call *skip-gram*. We then discuss alternative methods to assigning news sentiment that we use in the paper.

[6] We use a list of stop-words provided by the NLTK Python package.

A. Skip-gram

We train our model on article titles (headlines) from the PR data. We have 213,186 article titles with annotated sentiment scores using integers from −4 (the most negative sentiment score) to +4 (the most positive sentiment score). Titles of articles are short: they typically consist of one "sentence" and occasionally two. Accordingly, we apply our model to sentence–level data. When a title has two sentences, we assign the same sentiment score to both sentences and create a new data entry for the second sentence.

As is standard in the literature, we represent every sentence by a vector of individual word counts. However, we count not only single words but also two-word combinations. Specifically, we count word combinations that satisfy the following criteria.

1. Both words must be in English, according to the *2of12inf English dictionary*,[7] which means that both words are used in the English language.
2. At least one word in every two-word combination must appear in a list of 15,330 words that could have meaning in a financial context; we created this list by (a) using the Harvard General Inquirer system with relevant keywords (Econ, Legal, Positiv, Negativ, Strong, Weak, Active, Passive, Ovrst, Undrst, Persist, Quality, Quant) and (b) incorporating the LM dictionary's positive and negative word list.
3. To ensure that the word list can be used for other industries and does not "overfit" with respect to automotive and financial news, each focal word or word combination (irrespective of order) must appear in both the auto and finance data at least p times;

[7] See http://wordlist.aspell.net/12dicts-readme-r4

this criterion is intended to guarantee that there will be enough data to estimate coefficients for features, and *p* can therefore be viewed as a hyperparameter that needs to be estimated.

We estimate the sentiment charge of each word and word combination by regressing the human-coded sentiment score on the counts of our (filtered) single words and word combinations. Because the *Sentiment score* variable is coded as integers from −4 to +4, we use an ordered logit model. Our linear logistic model was trained using one Intel i7 CPU, requiring one afternoon.

More specifically, we use the threshold-based ordered logit model proposed by Rennie and Srebro (2005):

$$J(w) = \sum_{i=1}^{N}(\sum_{l=1}^{9} f(s(l; y_i)(\theta_l - w_0 - w'x_i))) + \frac{\alpha}{2}|w|^2 \qquad (1)$$

$$s(l\,; y_i) = \begin{cases} -1 \; if \; l < y_i \\ +1 \; if \; l \geq y_i \end{cases}$$

$$f(z_i) = \log(1 + \; e^{-z_i})$$

We find the optimal thresholds (θ_l) for every sentiment category l from -4 to +4 as well as the sentiment charge of every word or word combination (w) by minimizing equation (1). The second term in equation (1) is a regularization parameter used to punish more complex models; higher values of α increase the strength of the L2 regularization. Hence the

regularization strength (α) and the minimum word occurrences per sector (p) are hyperparameters requiring estimation. Standard cross-validation techniques can be used to "tune" both of these parameters. We use stratified sampling but ensure that the training and validation data in each stratum maintain the same proportion of automotive and finance articles as in the original data. Tuning of the hyperparameters and each word/combination's sentiment charge is based solely on the validation set's human-coded sentiment score values.

Applying this approach to the PR data and imposing $p > 1$, yields 45,676 features (word and word combinations), of which 5,706 are single words, and 39,970 are word combinations. We then use the estimated hyperparameters and equation (1) to assess the sentiment charge of each feature. The estimated parameter γ is a vector that assigns a coefficient to every word and word combination in our filtered sample; this coefficient reflects the respective features' positive or negative sentiment charge.

Table 3 reports the list of highest sentiment-charged words and word combinations according to our method. Panel A of Table 3 reports the 30 most positive and negative single words. The list of words is intuitive. It also suggests a strong presence of journalistic jargon. The most negative single words are "scandal", "cheated", and "deception". The most positive single words are "safest", "perception", and "sweeps." Of the top 30 negative single words, 23 also appear in the LM word lists. Of the top 30 positive single words, 17 appear in the LM word lists.

Panel B of Table 3 reports the 30 most positive and negative adjacent and non-adjacent word combinations. Again, the list seems intuitive. The three most negative words are: "biggest loss", "takes a hit", and "record fine". The three most positive word combinations are: "found

not" (as in found not guilty), "reports best," and "not liable." One of the more curious word combinations is "hot water," which made the list of negative word combinations. It was used in the following context: "Fiat (car company) in hot water for failing to report vehicle deaths, injuries…" Once again, the word list shows a strong presence of journalistic jargon. We also note other word combinations that appear in our word list but would less likely be used in more formal texts, such as "hot seat," "come clean," and takes heat."

[Insert Table 3 here]

Once we have the list of words and the associated weights, we can use them to assign sentiment scores to any newspaper article. In this paper, we will use our method to assign sentiment scores to two different samples. In Section IV, we use PR news articles to analyze how well our method captures human-perceived sentiment scores out-of-sample. In Section V, we use newspaper articles about S&P 500 companies to test for the relationship between news sentiment and stock returns. We always apply text pre-processing detailed in Section II.D. Moreover, since we trained our model on sentence-level data, we always split the text of every article into sentences. We separate the article title from the main body by a full stop (as explained below, this applies only to Section V). We predict the sentiment score of each sentence using equation (1) together with the sentiment charge (w values) estimated in the training step. We aggregate sentence-level sentiment into article sentiment scores by averaging the sentiment scores of all sentences within an article.

C. Other Methods

For comparison, we use two versions of the conventional "bag-of-words" approach to determining news sentiment. We also use two state-of-the-art machine-learning models that recently gained substantial traction.

C.1 Bag-of-words

Following the extant literature, we use Loughran and McDonald (2011) list of positive and negative words. Loughran and McDonald (2011) adjusted the list of positive and negative words from the Harward dictionary to the financial domain. We use their dictionary to create two commonly used sentiment scores in the finance literature – namely: (i) the percentage of negative words, which we denote as *% Negative LM* and (ii) *Net-tone LM*, defined as the percentage of negative words minus the percentage of positive words.

C.2 OpenAI

Next, we apply the sentiment scoring algorithm developed by OpenAI (the company behind ChatGPT). The OpenAI states the algorithm is trained on Amazon reviews and achieves

state-of-the-art sentiment analysis accuracy.[8] The training of OpenAI's model spanned a month, utilizing the power of four NVIDIA Pascal GPUs.

We accessed OpenAI's sentiment scoring algorithm through their paid API. We preprocessed our data and isolated each sentence. Through HTTP POST requests to their API, we then sent each sentence for sentiment analysis. The algorithm responded with sentiment scores, with values ranging from negative -4 to positive +4, indicating sentiment strength. We extracted these scores from the API responses and linked them with the corresponding sentences in our dataset. We then aggregated sentence-level scores to the article level.

C.3 FinBERT

We also apply FinBERT, a language model designed explicitly for financial sentiment analysis (Araci, 2019). It is based on the Bidirectional Encoder Representations from Transformers (BERT) architecture. BERT is a widely used language model in natural language processing. The original BERT model was trained on a large corpus of general text data, while FinBERT is pre-trained on a domain-specific financial dataset, such as earnings call transcripts, SEC filings, and other relevant financial texts. The training took place on a server equipped with 4 NVIDIA Tesla P100 GPUs, with the entire process taking around two days.

[8] https://openai.com/research/unsupervised-sentiment-neuron.

We obtained all the codes necessary to implement the FinBERT model from the hugging face repository.[9] We fed our preprocessed sentence-level news data into FinBERT. The model returned sentiment scores for each sentence, with values of either -1, 0, or 1. We aggregated sentence-level scores to the article level.

C.4 Weighted LM

Finally, we consider a reduced version of our skip-gram model. In many ways, our model blends the traditional "bag of words" approach with contemporary machine-learning techniques. Specifically, there are two main differences with respect to the traditional “bag-of-words” approach: (i) instead of using a pre-defined dictionary of positive and negative words, we let the computer pick domain-specific words and word combinations, and (ii) instead of assigning equal importance to each word, we let the computer determine the weights of each word and word combination. To assess the relative importance of each difference, we now consider a restricted version of our model. Specifically, we restrict ourselves to the word list from the LM dictionary, but we let the computer assign weights to each word best to fit the news sentiment in our training sample. We call this method *Weighted LM*.

[9] https://huggingface.co/ProsusAI/finbert

IV. Result

We first test how well the different methods explain human-coded news tone. Next, we ask whether news-based sentiment predicts daily stock returns.

A. Explaining human-coded news tone

For news articles coded by PR, we used the article titles to train the computer. Now, we use the main bodies of newspaper articles to validate our approach. We match articles in the PR data set with text from the Factiva database. We create 11,885 unambiguous matches between Factiva texts and the PR articles. We use only the body of the text and exclude titles because the latter were used to estimate the model in the training stage. Using the model trained on titles, we assign a sentiment score, ranging from −4 to +4, to every sentence in the article. Taking the average across all sentences in the article gives the article-level overall sentiment score. We then compare this sentiment score to the human-coded sentiment score. We compute the human-coded news sentiment score at the article level as the average of news sentiment scores across all the article segments (except for the title).

For comparison, we also determine the sentiment scores of PR articles using "bag-of-words" approaches, OpenAI, and FinBERT. Figure 2 presents the distribution of sentiment scores for each method. The distribution for the skip-gram model exhibits remarkable similarity with the distribution of human-coded sentiment scores: it is bimodal, with two peaks at -2 and +2, just like in the case of the human-annotated sentiment scores. Moreover, most observations are bunched around +2, again very similar to the case of human-annotated sentiment scores.

[Insert Figure 2 here]

The distribution of sentiment scores for the weighted LM is also bi-modal, but it is disproportionately skewed toward a score of +2. None of the other methods reproduces the bimodal distribution of news sentiment observed in the human-coded data. In fact, all the alternative methods have a unimodal distribution with a pronounced mode centered at zero. This suggests that to reproduce the bimodality in the distribution of news sentiment scores, it is essential to train the computer on the domain-specific newspaper-based training sample.

Next, we regress the human-coded article-level sentiment score on the sentiment scores produced by different sentiment models. Panel A of Table 4 reports the results for univariate regressions. Panel B of Table 4 includes control variables (number of words in an article, words per sentence, and a dummy variable for a picture).

Our skip-gram sentiment metric is positively and statistically significantly correlated with human-coded news sentiment in the regressions with and without control variables. Sentiment scores produced by reduced skip-gram, OpenAI, and FinBERT also positively and significantly correlate with the human-annotated sentiment. The "bag-of-words" measures, by construction, are negatively related to human-coded sentiment scores. The weighted LM, again, is positively and significantly correlated with the human-coded news sentiment.

Our sentiment measure exhibits the highest explanatory power across all the measures. The adjusted R-squared in the univariate regression is 42%. This compares to the adjusted R-squared of 38% for the weighted LM. Among the methods that are not explicitly trained on newspaper data, OpenAI performs the best with the adjusted R-squared of 41%. FinBERT

explains 36% of the variation in human-perceived sentiment scores. Among the bag-of-words approaches, *Net-tone LM* is slightly better than the *%Negative LM* with an adjusted R-squared of 38%.

[Insert Table 4 here]

B. Predicting Stock Returns

Next, we test whether news sentiment predicts short-term stock equity returns. All news in our Factiva dataset is from the newspapers' morning editions and thus released in the early morning before the market opens. Accordingly, our news sentiment measures are based on the information from before the market opens.

We regress daily open–close stock return on news sentiment score while controlling for standard variables in the finance literature:

$$\frac{Close_{i,t}-Open_{i,t}}{Open_{i,t}} = \alpha + \beta News\ Sentiment_{i,t} + \lambda Controls_{i,t} + \varepsilon_{i,t},$$

where $\frac{Close_{i,t}-Open_{i,t}}{Open_{i,t}}$ is the intra-day open–close return of stock *i* on day *t* calculated using open and closing prices as reported by CRSP. *News Sentiment* is measured using our skip-gram method or any of the alternative methods. For a given method, we take the average across sentiment scores for all articles that mention company *i* on day *t* to derive the overall sentiment score for that company on that day. We standardize all sentiment scores by subtracting the

mean and dividing by the standard deviation of the same sentiment score variable over the previous 6 months.

We use several control variables. Fang and Peress (2009) show that the media attention that a company receives when an article is published leads to lower stock returns, irrespective of the article's sentiment. To account for the media attention effect, we control for the logarithm of the number of words in the articles published that day. To further separate the effect of news sentiment from news coverage, we restrict our empirical test to the cross-section of stock returns on company-day pairs when at least one article is published. Because we use open-close returns, we control for the previous overnight return (i.e., until the market opens). For the rest of the control variables, we follow Tetlock et al. (2008). That is, we control for the firm's most recent earnings announcement using the standardized unexpected earnings (SUE) variable, the logarithm of company size (as measured by market capitalization), the logarithm of the firm's book-to-market value, and the logarithm of share turnover during the preceding 12 months. We follow Tetlock et al. (2008) in regressing a firm's stock returns on the Fama–French (1993) factors using data from 252 days to 31 days before the observation date. We then include as control variables the α of this regression (FFAlpha$(-252, -31)$), as well as the differences of the model's predicted return from the actual return of the previous day (FFCAR$(-1, -1)$), of the day before the previous day (FFCAR$(-2, -2)$), and of the entire month (FFCAR$(-30, -3)$).

Stock returns are correlated in the cross-section, and newspaper sentiment is serially correlated. One of the reasons for serial correlation in sentiment is that important news is often republished on subsequent days, leading to a stale news effect (Tetlock 2011). To control for

time and firm trends, we double cluster standard errors at the trading day and company level, as in Petersen (2009).

[Insert Table 5 here]

Table 5 reports the results. Our news sentiment measure based on skip grams is positively and significantly associated with future stock returns with a *t*-statistic of 2.42. This suggests that news sentiment is important for stock price movements.

None of the alternative sentiment metrics predict returns at the 5 percent level of statistical significance. The *t*-statistic associated with FinBERT predictor is 1.36. OpenAI measure is insignificantly negatively related to future returns. Among the conventional bag-of-words sentiment measures, net-tone LM has a stronger association with future returns (*t*-statistic of -1.67) than the % of negative words (t-statistic of -1.42).

In the last column of Table 5, we report results for the *Weighted LM*, where we restrict the list of words to the LM dictionary, but we use our training sample to estimate weights for each word. We find that such news sentiment does not predict daily returns (*t*-statistic of 0.9). This is aligned with our previous analysis (Figure 2), and it confirms that domain-specific training and allowing for word combinations are essential for the analysis of news sentiment.

Word combinations allow us to account for the effect of negations. According to Table 3, negations of positive words, such as “not supported”, “not relief”, and “not worth,” are all in the top decile of features associated with human-coded negative sentiment. Using word combinations also allows us to capture meaning from structures that are incorrectly classified

when using single-word dictionaries. For instance, the sentence "the case against the company was dropped" would be erroneously assessed as highly negative since "dropped" and "against" are negative words according to the LM dictionary. Yet Table 3 shows that "dropped against" is actually among the top word combinations associated with a human-perceived positive tone. Finally, words that themselves have a neutral meaning (and so are excluded from traditional dictionaries) can – when combined with other words – carry a strong sentiment charge; examples include "record sales", "ends probe", "record fine", and "pays price".

C. Factor Portfolios

In the previous section, we showed that our news sentiment measure predicts stock returns in a panel regression. We now consider a robustness check, whereby we create a long-short portfolio by simultaneously buying high-sentiment score stocks and selling low-sentiment score stocks. As before, we calculate daily company-specific news sentiment by averaging news sentiment scores across articles in which the company is mentioned. Companies not mentioned in the news on a given day are excluded from the analysis on that day.

We create a daily long-short portfolio as follows. Before the market opens, we group stocks into those that are above or below the median based on their printed news sentiment that morning. For the "long" leg of our news sentiment strategy, we buy stocks that are above the median news sentiment at the market open and hold them until market closing. At the market close, we sell these stocks and buy the S&P 500 Index. We hold the market overnight until the

market opens the following day. We similarly construct the "short" leg by shorting stocks that are below the median news sentiment at the market open and hold them until the market closes, at which point we close that position and sell the S&P 500 Index overnight. We repeat this process daily and take the difference between returns on the long- and the short leg. Finally, we regress the returns of the long-short portfolios on the risk factors in the cross-section of stock returns.

We construct long-short portfolios using sentiment scores from different sentiment methods. For each method, we regress the corresponding long-short portfolio returns on either the market model or the Fama-French five-factor model. All t-statistics are adjusted for autocorrelation using Newey–West (1987) adjusted standard errors up to lag 5.[10]

Table 6 reports the results. Columns 1 and 2 report results using our primary skip-gram method to determine news sentiment. The alpha for the long-short portfolio is sizable and always statistically significant. The estimated alpha coefficient is 0.050 with a *t*-statistic of 2.46 when we control for the market risk factor and 2.48 when we control for all five risk factors. None of the risk factors are statistically significant. The daily estimate of 0.050 implies an annualized alpha of 12.5%.

Among the alternative methods to determine news sentiment, only Finbert and Net-tone LM produce significant alphas at the 10% level. For Finbert, alpha is 0.035 with a *t*-statistic of

[10] Results are *not* adjusted for transaction costs.

1.76. For Net-tone LM, alpha is -0.035 with a *t*-statistic of -1.71. For the rest of the methods, alphas are insignificant.

[Insert Table 6 here]

In Figure 3, we illustrate cumulative daily returns from investing into the long-short portfolio, separately for each sentiment method. For our skip-gram method, $1 invested in January 2007 grows to a little bit more than $3.5 at the end of 2017. For Finbert and Net-tone LM, $1 invested in January 2007 grows to slightly over $2 at the end of 2017. For % Negative LM and Weighted LM, $1 grows to around $1.5. For OpenAI, the cumulative return line is downward sloping.

[Insert Figure 3 here]

V. Conclusion

We propose a novel approach to measuring news sentiment. Our model blends the simplicity and reproducibility of "bag of words" approaches with contemporary machine-learning techniques while leveraging high-quality news-sentiment scoring data. Specifically, we use data on human-coded sentiment scores from over 200,000 newspaper titles to train the computer to pick words and word combinations that are most closely associated with news sentiment as perceived by people. Thereby, we construct a list of words and two-word

combinations (and their corresponding weights) that is easy to apply in any given news sentiment analysis.

We apply our method to measure sentiment in newspaper articles about S&P 500 companies. The resulting sentiment scores are distributed bi-modally – with one negative peak and one positive peak, similar to a bi-modal distribution of sentiment scores observed in the human-coded data. Our news sentiment measure also predicts daily stock returns in a cross-section of stocks. No existing popular approaches to measuring sentiment (e.g., bag-of-words methods, OpenAI, Finbert) generate bi-modality in sentiment scores. Our news sentiment measure also works much better than the alternatives in predicting stock returns.

Overall, our results show that news sentiment is vital for stock price movements, but, if we want to measure news sentiment, we need to train the computer on a news-based training sample. The computational resources we utilized were substantially less, at least a thousand times less than those used by OpenAI and Finbert. Nevertheless, thanks to our domain-specific training sample, our model surpassed the performance of these larger, and more resource-intensive models. Our analysis also shows that it is essential to include both single words and word combinations. Moreover, allowing the computer to choose domain-specific words and word combinations appears more important than allowing the computer to pick weights for words from the existing dictionaries.

References

Ahern, K.R. and Sosyura, D., 2015. Rumor has it: Sensationalism in financial media. *The Review of Financial Studies*, *28*(7), pp.2050-2093.

Antweiler, W. and Frank, M.Z., 2004. Is all that talk just noise? The information content of internet stock message boards. *The Journal of finance*, *59*(3), pp.1259-1294.

Campbell, Y.J, Grossman, S. J. and Wang, J., 1993. Trading Volume and Serial Correlation in Stock Returns. *The Quarterly Journal of Economics*, 108(4), pp.905-939

Carhart, M.M., 1997. On persistence in mutual fund performance. *The Journal of finance*, *52*(1), pp.57-82.

Fama, Eugen F. and French, K.R., 1993. Common risk factors in the returns on stocks and bonds. Journal of Financial Economics, 33(1), pp.3-56.

Fama, E.F. and French, K.R., 2015. A five-factor asset pricing model. Journal of Financial Economics, 116(1), pp.1-22.

Fama, Eugen F. and MacBeth, J., 1973. Risk, return, and equilibrium: empirical tests. Journal of Political Economy 81, pp.607–636.

Fang, L. and Peress, J., 2009. Media coverage and the cross-section of stock returns. The Journal of *Finance*, *64*(5), pp.2023-2052.

Garcia, D., 2013. Sentiment during recessions. *The Journal of Finance*, *68*(3), pp.1267-1300.

Gentzkow, M., Kelly, B. and Taddy, M., 2019. Text as data. *Journal of Economic Literature*, *57*(3), pp.535-74.

Golez, B. and Karapandza, R., 2022. Home-country media slant and cross-listed stocks. *Working paper*

Heston, S.L. and Sinha, N.R., 2017. News vs. sentiment: Predicting stock returns from news stories. *Financial Analysts Journal*, *73*(3), pp.67-83.

Jegadeesh, N. and Wu, D., 2013. Word power: A new approach for content analysis. *Journal of financial economics*, *110*(3), pp.712-729.

Jiang, F., Lee, J., Martin, X. and Zhou, G., 2019. Manager sentiment and stock returns. Journal of Financial Economics, 132(1), pp.126-149.

Ke, Z.T., Kelly, B.T. and Xiu, D., 2019. *Predicting returns with text data* (No. w26186). National Bureau of Economic Research.

Khatua, A., Khatua, A. and Cambria, E., 2019. A tale of two epidemics: Contextual Word2Vec for classifying twitter streams during outbreaks. *Information Processing & Management*, *56*(1), pp.247-257.

Loughran, T. and McDonald, B., 2011. When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *The Journal of Finance*, *66*(1), pp.35-65.

Loughran, T. and McDonald, B., 2016. Textual analysis in accounting and finance: A survey. *Journal of Accounting Research*, *54*(4), pp.1187-1230.

Newey, W.K. and West, K.D., 1987. A simple, Positive Semi-definite Heteroscedasticity and Autocorrelation Consistent Covariance Matrix. *Econometrica*, *55*(3), pp.703-708.

Petersen, M.A., 2009. Estimating standard errors in finance panel data sets: Comparing approaches. *The Review of Financial Studies*, *22*(1), pp.435-480.

Rennie, J. D., & Srebro, N. (2005, July). Loss functions for preference levels: Regression with discrete ordered labels. In *Proceedings of the IJCAI multidisciplinary workshop on advances in preference handling* (Vol. 1). Kluwer Norwell, MA.

Salton, G. and Buckley, C., 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, *24*(5), pp. 513-523.

Sehrawat, S., 2019. Learning Word Embeddings from 10-K Filings for Financial NLP Tasks. *Available at SSRN 3480902*.

Soo, C.K., 2018. Quantifying sentiment with news media across local housing markets. The Review of Financial Studies, 31(10), pp.3689-3719.

Stone, Philip J., Hunt, Earl B. 1963. A computer approach to content analysis: studies using the General Inquirer system. In: Proceedings of the May 21–23

Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. *The Journal of finance*, *62*(3), 1139-1168.

Tetlock, P.C., 2011. All the news that's fit to reprint: Do investors react to stale information?. *The Review of Financial Studies*, *24*(5), pp.1481-1512.

Tetlock, P. C., Saar-Tsechansky, M., & Macskassy, S. (2008). More than words: Quantifying language to measure firms' fundamentals. The Journal of Finance, 63(3), 1437-1467.

Yu, Y., Yang, Y., Huang, J. and Tan, Y., 2019. Emotions in Online Reviews and Product Sales: Unifying Empirical and Theoretical Perspectives. *Available at SSRN 3497884*.

Table I
Prime Research data: Summary statistics

This table reports summary statistics for the proprietary Prime Research data set. The data set includes 213,186 English-language articles published from January 2007 to December 2016 in the U.S., United Kingdom, Canada, and Australia. Panel A shows the summary statistics for full-length articles. Panel B shows summary statistics for the titles of these articles (which are later used for machine training).

Panel A: Full Articles

Year	# Articles	Company coverage	Articles per company		Human-coded sentiment score	
			Mean	Median	Mean	Median
2007	11217	168	118.21	7	0.93	1.5
2008	10547	205	89.31	5	0.31	0.33
2009	10660	170	100.01	6	0.41	0.66
2010	9466	183	80.76	4	0.77	1.71
2011	9429	205	77.49	6	0.51	1.05
2012	16470	213	117.98	5	0.44	1
2013	31050	195	203.90	6	0.84	2
2014	38331	232	208.73	4	0.47	1
2015	33619	215	195.90	6	0.34	1
2016	42397	278	201.89	7.5	0.36	0

Panel B: Titles

Year	# Titles	Company coverage	Titles per company		Human-coded sentiment score	
			Mean	Median	Mean	Median
2007	11217	134	96.68	7	1.01	2
2008	10547	159	77.65	4	0.56	2
2009	10660	129	92.19	5	0.50	2
2010	9466	135	76.62	5	0.73	2
2011	9429	156	65.81	6	0.55	2
2012	16470	172	104.61	8	0.45	2
2013	31050	171	196.28	7	0.89	2
2014	38331	190	214.54	4	0.48	2
2015	33619	193	185.09	5	0.33	1
2016	42397	230	195.39	8	0.41	0

Table II
Factiva data: Summary statistics

This table reports summary statistics for our news data for SP 500 companies. The data set includes all articles published by the New York Times, the Wall Street Journal, the Washington Post, and USA Today from January 2007 to December 2017 that mentioned any firm in the S&P 500 Index. Like Tetlock (2008), we include only articles that have more than 50 words and mention the name of the firm at least once within the title or the first 25 words. The filtering criteria for article inclusion are summarized in the Table VII in the Appendix.

Year	# Articles	Company coverage	Words per article		Words per sentence		Articles per company	
			Mean	Median	Mean	Median	Mean	Median
2007	7,630	376	567.229	463	22.653	22.441	23.66	7
2008	9,017	360	547.542	461	22.714	22.462	30.592	8
2009	7,242	349	549.935	475	22.472	22.25	24.59	7
2010	6,410	329	562.738	489	22.611	22.389	22.97	7
2011	6,119	300	594.861	529	22.411	22.15	24.097	6
2012	5,779	297	616.419	543	22.353	22.174	22.502	5
2013	5,741	281	612.839	544	22.432	22.105	22.993	5
2014	6,083	283	608.345	545	22.143	22	25.219	5
2015	5,567	287	580.731	509	22.342	22.163	23.477	6
2016	5,275	265	570.638	488	22.313	22.13	23.796	6
2017	4,294	242	612.515	511	22.427	22.161	22.161	5

Table III
Skip-gram word list

This table reports the sentiment charge for the 30 most positive and negative single words and word combinations from our Prime Research training sample. The sentiment charge for each word is generated using the skip-gram model with parameters that yield the highest precision in the validation sets. Panel A shows the 30 most positive and most negative single words according to this method, along with their corresponding sentiment charge; Panel B does the same for the 30 most positive and most negative word combinations.

	Panel A: Single words				**Panel B**: Word combinations			
Rank	Most positive	Weight	Most negative	Weight	Most positive	Weight	Most negative	Weight
1	safest	4.34	scandal	-6.43	found not (guilty)	4.94	biggest loss	-5.33
2	perfection	3.98	cheated	-6.25	reports best	3.81	takes hit	-5.13
3	sweeps	3.90	deception	-5.97	not liable	3.78	record fine	-5.10
4	best	3.82	recall	-5.50	home run	3.68	emissions case	-4.50
5	wins	3.68	cheating	-5.39	against dropped	3.52	back buy	-4.04
6	named	3.59	worse	-5.12	record sales	3.49	pays price	-3.95
7	longest	3.46	auditors	-4.89	record profit	3.45	hot water	-3.90
8	earns	3.33	cheat	-4.86	ends probe	3.29	crisis made	-3.82
9	tops	3.23	loss	-4.78	million sales	3.28	record loss	-3.81
10	brilliant	3.22	plunge	-4.71	scores big	3.22	let go	-3.80
11	winner	3.16	Toll	-4.60	profit quadruples	3.20	down percent	-3.77
12	highest	3.14	sues	-4.52	takes spot	3.18	list worst	-3.74
13	seconds	3.06	death	-4.46	face won	3.17	come clean	-3.70
14	honored	3.03	embarrassing	-4.43	million sold	3.16	profit plummets	-3.69
15	selected	2.96	violated	-4.43	number one	3.14	offers settlement	-3.66
16	crowned	2.89	deceit	-4.39	court dismisses	3.13	scale back	-3.65
17	clears	2.78	jailed	-4.34	cost less	3.08	hot seat	-3.63
18	fantastic	2.76	falsified	-4.32	record quarter	3.01	worst ever	-3.60
19	gravity	2.75	writedown	-4.32	judge dismisses	3.00	failing report	-3.60
20	voted	2.71	poor	-4.25	leads group	2.96	takes heat	-3.56
21	awards	2.71	arrested	-4.22	still top	2.96	low shares	-3.53
22	excellence	2.66	sued	-4.22	record profit	2.91	profits down	-3.53
23	ever	2.63	falls	-4.21	takes top	2.91	not worth	-3.53
24	heaven	2.63	faulty	-4.21	judge cuts	2.91	pressure grows	-3.51
25	greatest	2.62	hacked	-4.20	finds not	2.85	too high	-3.51
26	heal	2.61	Sue	-4.19	third year	2.82	clears hurdle	-3.50
27	helps	2.60	covering	-4.18	fix loss	2.79	grace fall	-3.46
28	achieves	2.60	halts	-4.14	profit triples	2.79	billion settlement	-3.45
29	applaud	2.57	taint	-4.14	against dismissed	2.77	lowest years	-3.43
30	roar	2.52	manipulated	-4.13	top selling	2.75	cover up	-3.42

Table IV
Explaining human-coded sentiment scores out-of-sample: Comparing methods

This table reports results for regressing the *Human-coded sentiment score* on the sentiment score produced by different sentiment models. The analysis is conducted at the article level. In column [1], we determine sentiment using our *Skip-gram model*. In columns [2] and [3], we report the results for sentiment scores based on *OpenAI* and *Finbert*. In columns [4] and [5], we report the results for the two versions of the traditional "bag-of-words" approaches based on the LM dictionary, the *Percent Negative LM* and *Net-tone LM*. In column [6], we report results for the restricted skip-gram, whereby we use the LM dictionary of positive and negative words but let the machine determine the weight (sentiment charge) for each word, the *Weighted LM*. Panel A reports univariate regressions of human-coded news sentiment on the sentiment determined by each method. In Panel B, we add control variables for the number of words in the article, the number of words per sentence, and a dummy for whether the article includes a photo. In parentheses below the estimated coefficients are t-statistics with standard errors clustered at the source-month level. The sample period is from January 2007 to December 2016.

Panel A

	Dependent variable: Human-coded sentiment score					
	(1)	(2)	(3)	(4)	(5)	(6)
Skip-gram	1.326^{***}					
	$t = 25.672$					
OpenAI		1.991^{***}				
		$t = 20.365$				
Finbert			4.336^{***}			
			$t = 15.204$			
% Negative LM				-0.532^{***}		
				$t = -25.026$		
Net-tone LM					-46.603^{***}	
					$t = -29.595$	
Weighted LM						1.284^{***}
						$t = 23.575$
Constant	-0.247^{***}	-0.301^{***}	0.598^{***}	1.646^{***}	1.060^{***}	-0.947^{***}
	$t = -3.328$	$t = -2.602$	$t = 9.450$	$t = 16.849$	$t = 16.567$	$t = -13.110$
Error clustering	Source+Month	Source+Month	Source+Month	Source+Month	Source+Month	Source+Month
Observations	11,887	9,987	11,885	11,887	11,887	11,887
Adjusted R^2	0.415	0.408	0.360	0.350	0.369	0.377

Panel B

	Dependent variable: Human-coded sentiment score					
	(1)	(2)	(3)	(4)	(5)	(6)
Skip-gram	1.339^{***}					
	$t = 26.855$					
OpenAI		1.959^{***}				
		$t = 25.542$				
Finbert			4.236^{***}			
			$t = 16.011$			
% Negative LM				-0.520^{***}		
				$t = -24.012$		
Net-tone LM					-45.773^{***}	
					$t = -28.673$	
Weighted LM						1.278^{***}
						$t = 24.432$
log(# Words)	-0.091^{***}	0.011	0.031	-0.069^{*}	-0.060	-0.148^{***}
	$t = -4.802$	$t = 0.211$	$t = 0.620$	$t = -1.908$	$t = -1.346$	$t = -3.868$
Words per sentence	0.012	-0.035^{***}	-0.019^{**}	-0.017^{***}	-0.008^{*}	0.004
	$t = 1.625$	$t = -4.790$	$t = -2.508$	$t = -2.697$	$t = -1.925$	$t = 0.842$
Picture included	0.009	0.020	0.056^{***}	0.044^{***}	0.039^{***}	0.041^{***}
	$t = 1.008$	$t = 1.228$	$t = 4.162$	$t = 5.141$	$t = 3.825$	$t = 4.539$
Constant	0.103	0.236	0.671^{**}	2.309^{***}	1.535^{***}	-0.117
	$t = 0.648$	$t = 0.595$	$t = 2.328$	$t = 9.126$	$t = 5.252$	$t = -0.633$
Error clustering	Source+Month	Source+Month	Source+Month	Source+Month	Source+Month	Source+Month
Observations	11,887	9,987	11,885	11,887	11,887	11,887
Adjusted R^2	0.417	0.416	0.368	0.354	0.372	0.380

Table V
Predicting stock returns

This table reports results from the following regression:

$$\frac{Close_{i,t} - Open_{i,t}}{Open_{i,t}} = \alpha + \beta NewsSentiment_{i,t} + \lambda Controls_{i,t} + \varepsilon_{i,t},$$

where $\frac{Close_{i,t} - Open_{i,t}}{Open_{i,t}}$ is the intra-day open-close return of stock i on day t calculated using opening and closing prices as reported by CRSP. News sentiment is measured differently in each column of the table. We assign news sentiment on an article level and then average sentiment scores of all articles for firm i on day t to derive the sentiment score for that firm on that day. We use different textual analysis methods. In column [1], we determine sentiment using our *Skip-gram model*. In columns [2] and [3], we report the results for sentiment scores based on *OpenAI* and *Finbert*. In columns [4] and [5], we report the results for the two versions of the traditional "bag-of-words" approaches based on the LM dictionary, the *Percent Negative LM* and *Net-tone LM*. In column [6], we report results for the restricted skip-gram, whereby we use the LM dictionary of positive and negative words but let the machine determine the weight (sentiment charge) for each word, the *Weighted LM*. We standardize all sentiment variables for firm i on day t by subtracting the mean and then dividing by the standard deviation of the same sentiment variable over the past six months. The regression includes several control variables for past returns: the overnight return from yesterday's close to today's open, cumulative abnormal returns above a Fama and French three-factor model's predicted return on the previous day *FFCAR(-1, -1)*; two days ago *FFCAR(-2, -2)*; of the 30 days leading up to the day of the observation, *FFCAR(-30, -3)*; and the actual alpha of the Fama and French three-factor model estimated over the past year, *FFAlpha(-252, -31)*. Additional control variables include standardized unexpected earnings (SUE), market cap (Size), book-to-market value (B/M), share turnover in the previous year, and the number of words within all articles about firm i on a given day t. We closely follow the identification of Tetlock (2008). In parentheses below the estimated coefficients are t-statistics with errors clustered at the trading day and firm level (after Petersen (2009)). The sample period is from January 2007 to December 2017.

	Dependent variable: Intra-daily open-close returns					
	(1)	(2)	(3)	(4)	(5)	(6)
Skip-gram	0.029**					
	$t = 2.424$					
OpenAI		−0.011				
		$t = -0.736$				
Finbert			0.018			
			$t = 1.361$			
% Negative LM				−0.017		
				$t = -1.416$		
Net-tone LM					−0.020*	
					$t = -1.667$	
Weighted LM						0.012
						$t = 0.958$
Overnight Return	−0.587*	−0.587*	−0.586*	−0.584*	−0.585*	−0.584*
	$t = -1.864$	$t = -1.879$	$t = -1.860$	$t = -1.859$	$t = -1.860$	$t = -1.859$
FFCAR(-1,-1)	−0.737	−0.844	−0.744	−0.703	−0.715	−0.697
	$t = -0.430$	$t = -0.477$	$t = -0.432$	$t = -0.410$	$t = -0.417$	$t = -0.407$
FFCAR(-2,-2)	−3.101	−3.213	−3.102	−3.091	−3.096	−3.091
	$t = -1.004$	$t = -0.959$	$t = -1.004$	$t = -1.001$	$t = -1.002$	$t = -1.001$
FFAlpha(-252,-31)	6.422	−3.620	6.607	6.694	6.630	6.754
	$t = 0.225$	$t = -0.124$	$t = 0.231$	$t = 0.234$	$t = 0.232$	$t = 0.236$
SUE	0.059	0.053	0.059	0.059	0.059	0.059
	$t = 0.395$	$t = 0.359$	$t = 0.396$	$t = 0.397$	$t = 0.397$	$t = 0.396$
log(size)	0.121	0.129	0.122	0.122	0.122	0.122
	$t = 1.486$	$t = 1.512$	$t = 1.487$	$t = 1.490$	$t = 1.489$	$t = 1.491$
log(BM)	−0.030	−0.050	−0.030	−0.030	−0.030	−0.030
	$t = -0.777$	$t = -1.260$	$t = -0.780$	$t = -0.786$	$t = -0.786$	$t = -0.784$
log(Share turnover)	−0.049	−0.049	−0.049	−0.049	−0.049	−0.049
	$t = -0.819$	$t = -0.783$	$t = -0.821$	$t = -0.818$	$t = -0.819$	$t = -0.819$
log(# words)	−0.031	−0.031	−0.032	−0.032	−0.032	−0.032
	$t = -1.253$	$t = -1.170$	$t = -1.270$	$t = -1.271$	$t = -1.297$	$t = -1.271$
Error clustering	Date	Date	Date	Date	Date	Date
Fixed Effects	Firm	Firm	Firm	Firm	Firm	Firm
Observations	40,710	36,990	40,708	40,710	40,710	40,710
Adjusted R^2	0.011	0.013	0.011	0.011	0.011	0.011

Table VI
Risk factor: News sentiment

This table presents the results of regressing daily long-short portfolio returns on the risk factors. All portfolios are rebalanced daily. Every morning, stocks are grouped into two equally weighted portfolios based on their firm-specific, standardized news sentiment. We go long the "high" sentiment portfolio and short the "low" sentiment portfolio. The portfolios are held until the market close when they are liquidated, and the proceeds are invested in the S&P 500 Index overnight. The time series of each long-short portfolio is regressed on the market risk factor or the Fama and French five-factor model. In columns [1-2], we determine sentiment using our *Skip-gram model*. In columns [3-4] and [5-6], we report the results for sentiment scores based on the *OpenAI* and *Finbert*. In columns [7-8] and [9-10], we report the results for the two versions of the traditional "bag-of-words" approach based on the LM dictionary, the *Percent Negative LM* and *Net-tone LM*. In columns [11-12], we report results for the restricted skip-gram, whereby we use the LM dictionary of positive and negative words but let the machine determine the weight (sentiment charge) for each word, the *Weighted LM*. In parentheses below the estimated coefficients are Newey-West (1987) *t*-statistics with five lags. The sample period is from January 2007 to December 2017.

	Dependent variable:											
	Long-short Returns											
	Skip-gram		OpenAI		Finbert		% Negative LM		Net-tone LM		Weighted LM	
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
Alpha	0.050**	0.050**	−0.024	−0.024	0.034*	0.035*	−0.025	−0.026	−0.035*	−0.035*	0.015	0.015
	t = 2.460	t = 2.476	t = −1.203	t = −1.195	t = 1.704	t = 1.756	t = −1.259	t = −1.287	t = −1.700	t = −1.714	t = 0.681	t = 0.702
Mkt-Rf	0.013	−0.007	−0.013	0.014	0.003	−0.010	0.025	0.030	0.028	0.030	−0.003	−0.015
	t = 0.632	t = −0.276	t = −0.350	t = 0.490	t = 0.102	t = −0.346	t = 0.677	t = 0.742	t = 0.990	t = 0.894	t = −0.087	t = −0.425
SMB		0.062		0.016		0.076		−0.005		−0.049		0.022
		t = 0.800		t = 0.305		t = 1.260		t = −0.079		t = −0.803		t = 0.340
HML		0.048		−0.136		0.010		−0.002		0.029		0.032
		t = 0.922		t = −1.243		t = 0.170		t = −0.024		t = 0.425		t = 0.432
RMW		0.009		−0.132		−0.043		0.028		0.039		−0.004
		t = 0.136		t = −1.456		t = −0.478		t = 0.304		t = 0.446		t = −0.054
CMA		−0.084		0.417*		0.098		0.010		−0.069		−0.054
		t = −0.701		t = 1.839		t = 0.997		t = 0.069		t = −0.559		t = −0.375
Newey West SE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Observations	2,765	2,765	2,764	2,764	2,765	2,765	2,765	2,765	2,765	2,765	2,764	2,764
R^2	0.0003	0.002	0.0002	0.013	0.00001	0.003	0.001	0.001	0.001	0.003	0.00001	0.001

Figure 1. Distribution of human-coded sentiment scores

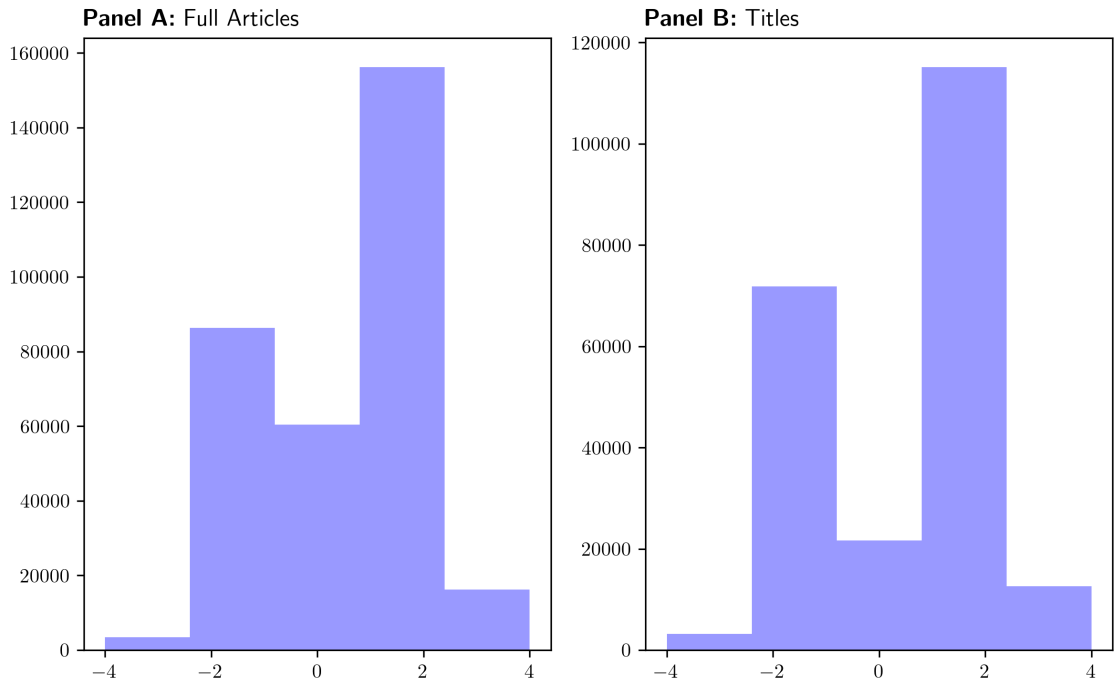


Figure 1. This figure presents the distributions of human-coded sentiment scores from the Prime Research dataset. Panel A plots the distribution of sentiment scores for the full-length articles. Panel B plots the distribution of sentiment scores for the article titles. Both panels contain 213,186 observations. The sample period is from January 2007 to December 2016.

Figure 2. In-sample distribution of sentiment scores: Comparing methods

This figure plots the distribution of sentiment scores at the sentence level. The data cover news about S&P 500 companies published by the New York Times, the Wall Street Journal, the Washington Post, and USA Today from January 2007 to December 2017 that we linked to Prime Research data. Histograms are based on *Human-coded sentiment scores*, our *Skip-gram model*, *OpenAI*, *Finbert*, *Percent Negative LM*, *Net-tone LM*, and *Weighted LM*.

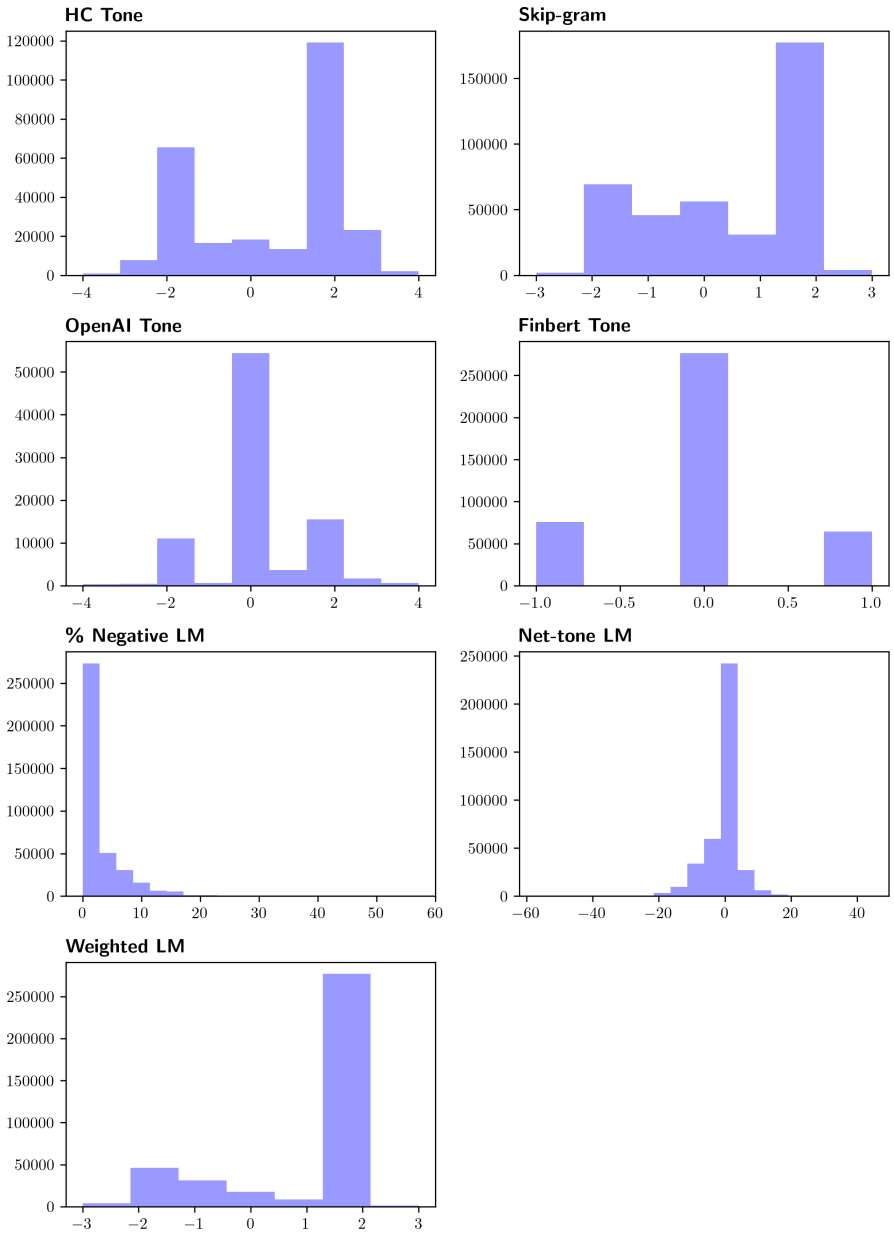
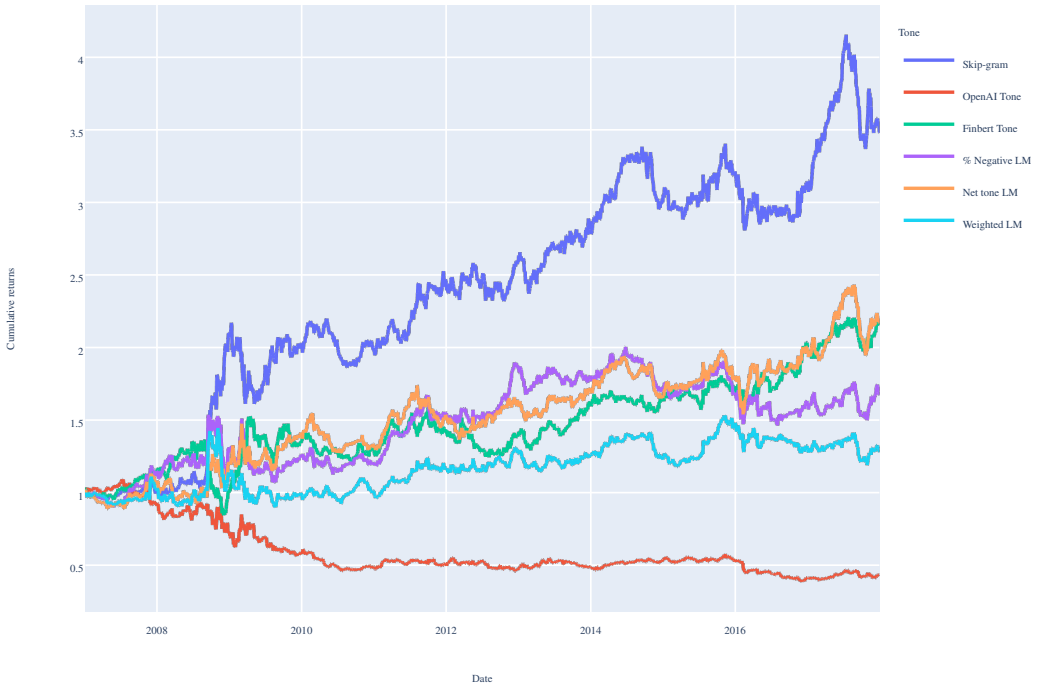


Figure 3. Cumulative portfolio returns

This figure plots the cumulative returns for long-short portfolios from Table VI. Dark blue line presents the cumulative returns for the strategy based on our *Skip-gram model*. Red and green lines present the cumulative returns for the strategies based on *OpenAI* and *Finbert*. Purple and orange lines present the cumulative returns for the strategies based on the traditional "bag-of-words" approaches (the *Percent Negative LM* and *Net-tone LM*). The light blue line presents the cumulative returns for the strategy based on the restricted skip-gram, whereby we use the LM dictionary of positive and negative words but let the machine determine the weight (sentiment charge) for each word, the *Weighted LM*. The sample period is from January 2007 to December 2017.



I. Appendix

Table VII
Filtering Factiva News

This table shows the extent to which each filter that we apply to Factiva news data reduces the number of observations in our sample. The sample consists of all newspaper articles $\times$ firm pairs published by the New York Times, the Wall Street Journal, the Washington Post, and USA Today from January 2007 to December 2017 that mention any of the S&P 500 firms at least once.

Filter	Observations removed	Observations remaining
Full Factiva sample (unfiltered)	NA	200,642
Tetlock, Saar-Tsechansky and Macskassy (2008) article matching	106,322	94,310
Remove articles with common titles that refer to tech reviews, sports, movie critics, home development, etc.	2,277	92,033
Remove articles whose sum of legal and economics/finance words amount to less than five percent of the total word count	8,698	83,335