Taylor & Francis
Taylor & Francis Group

Journal of Business Analytics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tjba20

Induction of a sentiment dictionary for financial analyst communication: a data-driven approach balancing machine learning and human intuition

Matthias Palmer, Jan Roeder & Jan Muntermann

To cite this article: Matthias Palmer, Jan Roeder & Jan Muntermann (2021): Induction of a sentiment dictionary for financial analyst communication: a data-driven approach balancing machine learning and human intuition, Journal of Business Analytics, DOI: 10.1080/2573234X.2021.1955022

To link to this article: https://doi.org/10.1080/2573234X.2021.1955022

JOURNAL OF BUSINESS ANALYTICS
https://doi.org/10.1080/2573234X.2021.1955022

Induction of a sentiment dictionary for financial analyst communication: a data-driven approach balancing machine learning and human intuition

Matthias Palmer, Jan Roeder and Jan Muntermann

Chair of Electronic Finance and Digital Markets, University of Goettingen, Goettingen, Germany

ABSTRACT

While sentiment dictionaries are easy to apply and provide reproducible results, they often exhibit inferior classification performance compared to machine learning approaches trained for specific application domains. Nevertheless, both approaches typically require manual data analysis. This paper develops a domain-specific dictionary using regularised linear models drawing from textual reports of financial analysts. The first evaluation step demonstrates that the developed financial analyst dictionary can explain cumulative abnormal stock returns related to earnings events more accurately compared to other finance-related dictionaries and sentiment classifiers. In a second step, the approaches are compared using manually annotated sentiment. The financial analyst dictionary is more accurate than other dictionary-based approaches, although it cannot compete with a pre-trained deep learning sentiment classifier. While we show that the proposed approach is suited for texts of financial analysts, it can be applied to other use cases. The approach realises context specificity while reducing extensive manual data analysis.

ARTICLE HISTORY
Received 27 November 2020
Accepted 6 July 2021

KEYWORDS
Sentiment analysis; domain-specific sentiment dictionary; dictionary induction; financial analysts; analyst reports

1. Introduction

Researchers and practitioners use sentiment dictionaries for the automated content analysis of text data. Sentiment dictionaries can detect positive or negative sentiment but also uncertainty, anger, and related emotions in a scalable fashion. For example, they can be used to study corporate disclosures or analyse the reaction of market participants to news events. Several literature reviews examine sentiment dictionaries and argue for an intensified domain-specific development (Kearney & Liu, 2014; Loughran & McDonald, 2016; Xing et al., 2018). Sentiment dictionaries contain polarity words, i.e., words indicating a positive or negative tone. In this context, domain-specific dictionaries are tailored to a particular topic or profession (Loughran & McDonald, 2015). Nonetheless, general-purpose dictionaries are still used in many studies. Commonly, there is no domain-specific dictionary readily available for the specific research context and its construction typically requires extensive efforts through manual data analysis (Loughran & McDonald, 2016). The potential complexity of creating a dictionary is further illustrated by Cambria et al. (2017). They identify sentiment analysis as a multi-layered application field that requires differentiated consideration, especially regarding machine learning-based approaches. In these approaches, statistical models are trained using large amounts of domain-specific data to classify text (Liu, 2012). By comparing machine learning and dictionary-based sentiment measures, Henry and Leone (2015) observe only slight differences between the results and therefore favour dictionary-based approaches, which they find easier to interpret and apply. Loughran and McDonald (2016) emphasise the potential drawbacks of black-box algorithms, as their opaqueness may overshadow their added value.

Determining sentiment in the texts of financial analysts is a common use case. Finance research demonstrated that analyst reports' sentiment plays an important role in interpreting company-related publications (Twedt & Rees, 2012). Huang et al. (2014) show that texts of analyst reports help to interpret quantitative capital market data and qualitative company disclosures using machine-learning methods. Moreover, Huang et al. (2017) show that the value of conference call interpretations by financial analysts is higher when analysts use their analyst-specific language, and markets react stronger when the analyst reports' sentiment values are particularly positive. To further improve the sentiment recognition in analysts' texts, Yang et al. (2020) train the BERT language model (Devlin et al., 2019) using large collections of corporate reports, conference calls, and analyst reports. The training of the BERT language model creates a pretrained analyst-specific sentiment classifier (FinBERT), which can be used for various data classification and representation tasks in the financial domain. Due to its pre-training, improved

CONTACT Matthias Palmer matthias.palmer@uni-goettingen.de Chair of Electronic Finance and Digital Markets, University of Goettingen, Goettingen, Germany

performance specifically for finance-related data is achieved and results in superior classification accuracy compared to existing approaches.

Jegadeesh and Wu (2013) and Pröllochs et al. (2015) are among the first to present approaches for the automated development of domain-specific sentiment dictionaries in the finance domain. Both papers combine stock price returns and domain-specific texts to assign different weightings to words. Jegadeesh and Wu (2013) include words solely from existing polarity word lists, while Pröllochs et al. (2015) use the analysed texts' most relevant words. The two approaches mentioned above stand in contrast to the FinBERT model since they are explicitly designed to generate easy-to-use word lists instead of a complex machine learning model. However, it must be noted that this also involves accepting performance losses in the sentiment classification compared to machine-learning approaches, e.g., the FinBERT model.

We are encouraged by the work of Henry and Leone (2015), who explicitly advocate for the induction of a sentiment dictionary for analyst reports. Therefore, we combine the approaches of Jegadeesh and Wu (2013) and Pröllochs et al. (2015) by identifying domain-specific words and extending a general-purpose sentiment dictionary to induce a financial analyst dictionary (hereinafter referred to as the FA dictionary). The polarity term identification is based on relating the textual analyst reports, which tend to cluster around earnings releases, to the related company's stock returns associated with said earnings releases. The widely used event study methodology supports extracting the abnormal portion of stock returns from the more general market trend. Thereby, event studies allow to assess the impact of the earnings releases (i.e., the events) on the related company's value. This procedure helps to identify the most relevant words in analyst reports effectively. In order to ensure that the dictionary can be applied even to short text passages, we deem it necessary to evaluate the performance in a setting where the word occurrences for the whole corpus are not known upfront. Furthermore, we do not limit the FA dictionary to single words but also include identified phrases. In light of our outlined approach, we pose the following research questions:

***RQ 1**: How to effectively develop finance-specific sentiment dictionaries by extending more general-purpose dictionaries?*

Subsequently, we evaluate the developed FA dictionary in explaining stock market returns and investigate how it compares to alternative dictionaries when it comes to manually labelled data. Consequently, we raise the following second research question:

***RQ 2**: How does a semi-automatically generated domain-specific sentiment dictionary perform compared to less-specific dictionaries in the context of analyst communication?*

The paper is structured as follows. Section 2 begins with the basics of domain-specific sentiment analysis and presents a brief introduction to financial analysts. In section 3, we set forth the methodological background for sentiment dictionary induction and event study analysis. Section 4 contains the description of the dataset. Section 5 explains the word polarisation setup, and section 6 evaluates the FA dictionary compared to existing sentiment classification approaches. Section 7 discusses the results and section 8 contains a conclusion in addition to future research opportunities.

2. Related literature

2.1. Sentiment analysis

Sentiment analysis measures the emotional tendencies of a text. It can be determined whether a sentence is objective or subjective and whether the text expresses a positive, neutral, or negative sentiment. Aspect-based sentiment analysis can help to analyse to which products or topics expressed opinions may refer. As a subcategory of Natural Language Processing (NLP), sentiment analysis can automatically analyse large quantities of unstructured texts. At its core, sentiment analysis is an intuitive task. Still, the more the sentiment analysis methods are adapted to the text's topic or the author's background, the more involved they can become. For example, the word meanings in different domains can be quite different. Li (2010b), Kearney and Liu (2014), and Das (2014) provide literature reviews on textual analysis for diverse fields of application and methods. Sentiment analysis can be roughly categorised into two different analytical approaches: machine learning-based and dictionary-based sentiment analysis (Kearney & Liu, 2014).

Machine learning-based approaches use supervised learning techniques that require labelled (i.e., annotated) training and test datasets. Different models, e.g., linear, rule-based, or probabilistic, are trained and subsequently tested based on the dataset. The data labelling can be carried out manually at the sentence, paragraph, or document level. Antweiler and Frank (2004) use this procedure to evaluate Internet bulletin board messages, Das and Chen (2007) analyse stock message board postings, and Li (2010a) measures the sentiment in management discussions and analysis disclosures. Huang et al. (2014) label sentences in analyst reports and train a Naïve Bayes classifier. Naturally, there is an inherent risk of incorrect labelling. Jegadeesh and Wu (2013) train a sentiment classifier based on stock market reactions to 10-K reports and avoid sentences' subjective labelling. A sentiment

classification considerably depends on the training dataset. Structural biases inherent to a dataset might be reflected in the resulting model if no measures are taken to address this issue. Then, a sentiment model might extract features that are proxies for other measures, e.g., the features might be just rough dummies for industries (Loughran & McDonald, 2016).

Dictionary-based sentiment analysis is based on word lists. Utilising dictionaries is appealing since word lists are easy to understand and more intuitive to handle than advanced model-based learning approaches. The word occurrences of the different categories (commonly positive and negative) are counted in the text, and a sentiment score is calculated from the ratio of the polarity words. Word lists can also be used for aspects such as uncertainty or objectivity. However, this is less common in finance research (Loughran & McDonald, 2016). Sentiment dictionaries can be applied to different datasets relatively quickly and do not need to be re-trained. Using the same word list and the same data preparation steps will provide the same result. There are somewhat general dictionaries developed for social sciences, e.g., General Inquirer (Stone et al., 1966) and DICTION (Hart, 2000). Their creation process often contains deductive reasoning, while also inductive components such as statistical word occurrences are used. Furthermore, specific dictionaries for product reviews (Hu & Liu, 2004) or microblogging data analysis (Oliveira et al., 2017) exist. Henry (2008) developed a dictionary for earnings announcements with 105 positive and 85 negative words using a Thesaurus-based approach. A disadvantage of this dictionary is the non-exhaustive listing of negative words, as outlined by Loughran and McDonald (2011). Hence, Loughran and McDonald (2011) created a dictionary using word counts of 10-K filings. It contains 354 positive and 2,329 negative words. The latter two dictionaries are commonly used in finance research and practice. In this paper, we refer to them as the Henry and the LM dictionary. It has been shown that domain-specific dictionaries work better in their designated domain. For instance, almost three-fourths of the negative words from the non-domain-specific General Inquirer cannot be assigned to negative sentiment in a financial context (Loughran & McDonald, 2011). Other approaches successfully combine existing word lists (Demers & Vega, 2014; Rogers et al., 2011). Pröllochs et al. (2015) use a (mostly automated) regularised regression analysis to determine financial news disclosure's polar terms empirically. Furthermore, there are dictionaries for different languages, e.g., Chinese (Peng et al., 2017), Arabic (Mahyoub et al., 2014), and German (Remus et al., 2010), and also bilingual approaches (Lu et al., 2011).

2.2. Financial analysts and analyst reports

Financial analysts are industry experts who analyse companies to assess their prospects. For this purpose, analysts examine financial ratios, business models, management, the industry, and the overall economic situation. In this context, sell-side analysts try to communicate closely with company representatives to obtain new information and pass it on to their firm's clients (Soltes, 2014). To this end, sell-side analysts publish relevant information in analyst reports at regular intervals (Twedt & Rees, 2012) and support information discovery and interpretation (Chen et al., 2010). In these reports, analysts give general recommendations about whether a company's share should be bought and provide price targets for a specific period. Analysts offer concise assessments of the latest company developments and in-depth analyses of the business model or market development. Analyst reports are often written by a team of analysts headed by a senior analyst with several years of industry expertise. The word choice in these reports is consequently shaped by many finance and management-specific terms. Besides, there is a distinct writing style in the reports, which differs from corporate publications, public financial reporting, or management texts. For this reason, it appears imperative to develop a sentiment-dictionary that is specifically trained for the language of financial analysts (Henry & Leone, 2015).

3. Methodological background

3.1. Sentiment dictionary induction

In a comprehensive literature review, Mengelkamp (2017) highlights the increasing number of newly developed sentiment dictionaries within the last years. This review examines the development process steps according to which the different approaches for creating sentiment dictionaries can be categorised. These steps roughly correspond to the procedure that Pröllochs et al. (2015) carries out, which serves as an orientation in implementing our approach. A distinction is made between the initial *construction* and the *extension* of a sentiment dictionary in the literature. The *construction* step consists of three phases: words are *selected*, then *polarised*, and subsequently *evaluated*. In the course of a dictionary *extension*, features might be edited or newly added, e.g., synonyms, antonyms, or other mood indicators such as emoticons. In addition, the basic dataset can be

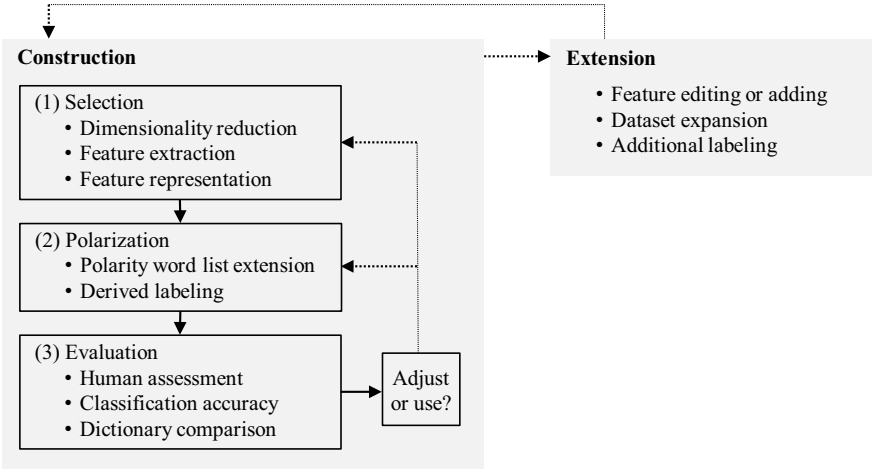

Figure 1. Sentiment dictionary induction process.

extended, or additional data labelling can be carried out. The sentiment dictionary induction process is illustrated in Figure 1.

In the *selection phase*, appropriate data sources need to be chosen (see section 4). These include existing sentiment dictionaries, adequate text corpora, the knowledge of native speakers, or lexical-semantic databases. Depending on the data selection, the data must be formatted further. After a *dimensionality reduction* of texts, e.g., removing stop words, irrelevant text parts, duplicates, and whitespace, the *feature extraction* follows, e.g., by utilising bag-of-words, N-grams (i.e., sequences of N adjacent words), or Part-of-Speech tagging (Nassirtoussi et al., 2014). The *feature representation* follows this. Here, a term-document matrix (TDM) is usually created using a binary, term frequency – inverse document frequency (tf–idf) or chi-squared approach. Depending on the text corpora used, it is possible to develop domain-specific dictionaries (Kearney & Liu, 2014).

In the *polarisation phase* (see section 5), it is determined whether a word is positive, neutral, or negative. The classification depends on the purpose of the dictionary. Previous studies mostly build on existing polarity lists (Loughran & McDonald, 2016). At the same time, word polarity can be derived from other text classifications. For example, product ratings or capital market returns may be used.

The *evaluation phase* (see section 6) can consist of three approaches that potentially complement each other (Mengelkamp, 2017). Firstly, native speakers can evaluate dictionaries. Secondly, labelled evaluation datasets can be utilised to assess classification accuracy. Such an evaluation dataset typically consists of unseen data, e.g., more recent data, that has not been used for the dictionary induction. A sensitivity analysis can show how the pre-processing, data representation, or classification needs to be adjusted depending on the accuracy. Thirdly, based on an evaluation dataset, a comparison can be made to other dictionaries belonging to a related domain.

3.2. Event study analysis

The event study methodology we describe in the following bridges the gap between capital market valuation effects and the textual analyst reports. It is possible to assign the capital market reaction to the analyst reports, thereby creating a labelled dataset, which is then used for the word polarisation later on. The return of a company's stock is compared to a reference market to identify abnormal capital market returns. An event study measures whether a previously defined event type leads to a change in a company's stock returns that would not have occurred without the event's new information. The following description of the event study builds primarily on MacKinlay (1997) and McWilliams and Siegel (1997). For company i and time t, equation (1) defines the abnormal return (AR) as:

$$AR_{i,t} = R_{i,t} - E\left(R_{i,t}|R_{m,t}\right) \tag{1}$$

$AR_{i,t}$ is the abnormal return, which is the difference between the actual return $R_{i,t}$ and the normal return $E\left(R_{i,t}|R_{m,t}\right)$. For the market model used in this study, $R_{m,t}$ is defined to be the market return (MacKinlay, 1997). As calculating the market return does not require additional parameters, the normal return is now specified in more detail. Equation (2) defines the market model for any security i as:

$$R_{i,t} = \alpha_i + \beta_i R_{m,t} + \varepsilon_{i,t} \tag{2}$$

$$E\left(\varepsilon_{i,t}\right) = 0 var\left(\varepsilon_{i,t}\right) = \sigma_\varepsilon^2 \tag{3}$$

$R_{i,t}$ is the return of security i in period t and $R_{m,t}$ the return of market m in period t. The ordinary least squares (OLS) method helps to estimate the intercept α_i, the systematic risk β_i, and σ_ε^2 using historical data,

which is constrained to the estimation window (McWilliams & Siegel, 1997). The model can then calculate abnormal returns, as described in (1) (McWilliams & Siegel, 1997). To characterise a specific event using *ARs*, they can be aggregated across the time dimension, which results in the cumulative abnormal return (*CAR*). This is typically done for a pre-defined event window (*EW*), starting at day EW_1 before the event date and ending at day EW_2 after the event date (also written as $[EW_1, EW_2]$). The *CAR* is defined as shown in equation (4):

$$CAR(EW_1, EW_2) = \sum_{t=EW_1}^{EW_2} AR_{i,t} \tag{4}$$

A hypothesis test can be conducted, assuming a normal distribution of the *CARs*. The expression in equation (5) describes the null hypothesis:

$$H_0 : CAR_i = 0 \tag{5}$$

The variance of *CAR* in equation (6) is asymptotically given by the variance of the residuals of the linear model for the estimation window, which is scaled by the event window size (MacKinlay, 1997):

$$\sigma_i^2(EW_1, EW_2) = (EW_2 - EW_1 + 1)\sigma_{\epsilon_i}^2 \tag{6}$$

Equation (7) shows the calculation of the *t*-value (Schimmer et al., 2014). For the market model case, the square root of the equation above specifies the fraction's denominator (MacKinlay, 1997).

$$t_{CAR} = \frac{CAR_i}{\sigma_i^2(EW_1, EW_2)^{\frac{1}{2}}} \tag{7}$$

4. Data selection, pre-processing, and representation

4.1. Analyst reports

We use a dataset that is based on the equity index Dow Jones Industrial Average (DJIA). The DJIA includes the 30 largest U.S. companies. Based on our observations of the Institutional Brokers' Estimate System (I/B/E/S), these 30 companies tend to be frequently covered by financial analysts. The dataset covers the period from 2009 to 2019, and companies that were present in the DJIA as of 12/31/2019 are analysed. One particularity is that the data quality for Dow, Inc. and its predecessors is not satisfactory due to the mergers and spin-offs. Therefore, the previous index constituent General Electric Co. is incorporated into the analysis. The data is obtained from Thomson Reuters Advanced Analytics. Non-broker research as classified by Thomson Reuters is removed before the data retrieval to ensure a consistent quality level of the research documents. The removal is necessary because the non-broker analyst reports tend to exhibit irregularities. For example, these documents are significantly longer or shorter compared to reports of regular brokers. Overall, this results in a total number of 76,951 analyst reports. In the following, we give detailed explanations for the steps in which the data is pre-processed and filtered (see Table 1 for an overview).

Table 1. Data selection and pre-processing steps for analyst reports.

Step	Short description	Remaining reports
	Available analyst reports	76,951
1	Transformation of PDF files to a structured representation	76,951
2	Filter by report titles	75,799
3	Filter by broker	70,508
4	Deletion of reoccurring text parts (boilerplate removal)	69,806
5	Filter by company event dates	14,709
6	Tokenisation, deletion of stop word, and custom-term removal	14,709
7	Phrase detection (bigrams)	14,709
8	Sentence extraction	14,709

Step 1: A manual analysis of the analyst reports shows that essential content is primarily located on the reports' first pages. Therefore, the following dictionary induction focuses on the first five pages. Since the reports are provided as PDF files, we must deal with a data format that is not suited for the reliable storage and subsequent extraction of semi-structured data. This issue is addressed by carefully transforming the data into a tabular structure. Each text block is now contained in a single cell. In this way, the basic structures and thematic sections of the individual documents are kept, which is advantageous for subsequent text processing. The text blocks, which are now each stored as a cell, can vary in appearance depending on the author and, accordingly, the structure and formatting of the document. The text blocks can consist of single headings as well as several long paragraphs. The text cells are filtered based on heuristics, such as a minimum number of words or the ratio of words to numbers. In the next step, special characters are removed. This process results in higher data quality, as complete sentences are kept while headings or tables are discarded.

Step 2: The reports' titles indicate whether they are regular reports or, for example, cover multiple companies. Reports, where the assignment based on the title is ambiguous, are removed. The dataset thus shrinks to 75,799 reports.

Step 3: We ensure that only brokers with a minimum number of published reports are included in the dataset in the next step. In this case, brokers that have published at least 20 reports during the observation period remain in the dataset. This step reduces the number of considered brokers from 288 to 118, which have published 74,979 reports. For the remaining brokers in the dataset, the reports were manually quality-checked on a random sample basis. Brokers publishing reports that were not manually generated, do not

contain analyst names, and do not contain price targets and recommendations have been discarded. This further data cleaning reduces the dataset to 102 brokers and 70,508 reports.

Step 4: To simplify the detection of recurring text segments, we standardise the following elements: numbers, dates, emails, and web addresses. In the following, we delete text parts (i.e., boilerplate) representing frequently recurring standard texts of brokers. In some cases, this can be true for entire reports. For each text cell in the dataset, the number of occurrences is determined. Text cells that occur at least five times are deleted. Furthermore, if a broker uses that exact wording for more than two companies within one year, the text cell is deleted. The number of text cells decreases from 2,550,716 to 1,146,740. This reduction reveals the high proportion of recurring text parts that could harm the dictionary creation. To deal with cases in which only minor text adjustments were made within a paragraph, the text cells are cut into paragraphs, and the aforementioned duplicate detection is repeated. These steps reduce the number of paragraphs from 1,514,358 to 1,114,484. Overall, the number of reports is 69,806 after this step.

Step 5: The following dictionary induction requires that only those reports are considered that were released during the ten days after the companies' quarterly earnings releases. Figure 2 shows the 14,709 analyst reports remaining in the final dataset for the ten-day time frame per quarter. It is important to filter after the previous processing steps to ensure that all text occurrences are present for duplicate detection.

Step 6: We group and merge the text cells at the analyst report level for further pre-processing and analysis. Numbers are filtered, and texts are transformed to lower case and tokenised. Then, stop words and words with less than three characters are deleted. Subsequently, the following texts get removed: date-related words, the companies' names, the corresponding tickers, and the analyst firms' names.

Step 7: Not only individual words but also word combinations characterise texts. Instead of evaluating all possible combinations of two particular words (bigrams or 2-grams as opposed to unigrams or 1-grams), we attempt to identify those that tend to occur with one another. For this purpose, Mikolov et al. (2013) propose a method that Řehůřek and Sojka (2010) implement. The basic intuition is that we want to identify words that often occur together in relation to the overall occurrence. For each combination of two successive words, $word_i$ and $word_j$, the bigram score is calculated, as shown in equation (8):

$$bigram\,score(w_i, w_j) = \frac{\left(n_{w_i w_j} - discount\right) * vocabulary}{n_{w_i} * n_{w_j}}, \quad (8)$$

where $n_{w_i w_j}$ is the number of co-occurrences in the corpus and *discount* the discounting coefficient. The *vocabulary* is the number of unique words in the corpus. It was added in the implementation of Řehůřek and Sojka (2010) to help pruning infrequently occurring words. n_{w_i} and n_{w_j} are the occurrences for each word. After calculating the bigram score, a user-specified threshold controls which phrases are included, as the method only incorporates phrases with a score larger than the threshold. To identify a suitable value, the *threshold* variable is initialised at 20, which is twice as high compared to the standard value in the used library. A manual review of the identified phrases shows that lower *thresholds* add sensible phrases. Hence, the value is decremented from 20 to 1 (from restrictive to inclusive) with a step size of 1. Afterwards, the resulting lists of bigrams and especially changes between these lists are reviewed manually. This analysis suggests that a threshold of 2 provides good word combinations.

Step 8: We also create a representation of the dataset in which we divide the reports into sentences. On

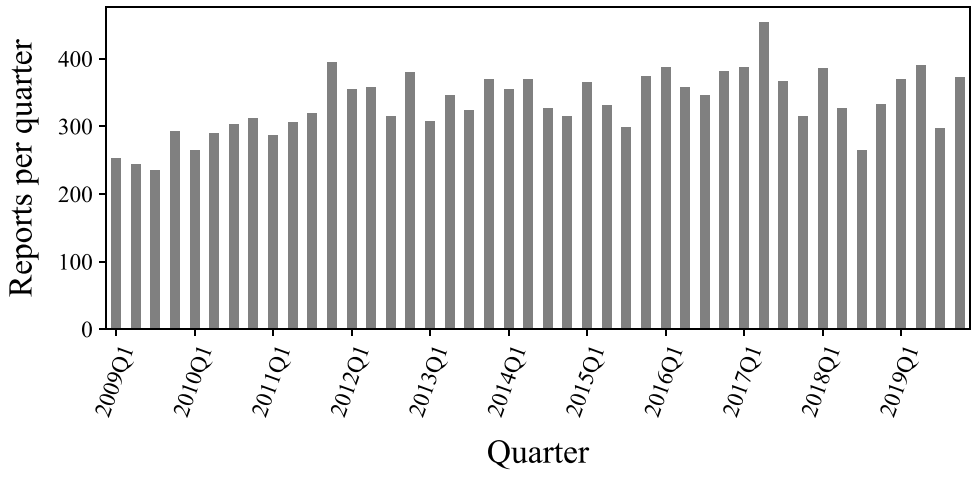

Figure 2. Number of analyst reports published up to ten days after an earnings release and included in the final dataset.

the one hand, this is necessary to apply the FinBERT classifier (Yang et al., 2020) to the reports and compare it to the FA dictionary. On the other hand, this allows us to label sentences manually and evaluate the FA dictionary's performance later on. To do this, we use the representation of the reports, which still contains all sentences in full. Sentences are recognised with the *NLTK (Natural Language Toolkit) Punkt Sentence Tokenizer*. A pre-trained English sentence model is used for this purpose. The sentence boundary detection is further improved by training a new analyst-specific sentence model using the dataset. The model learns typical abbreviations, sentence starters, collocations (with a period at the end of the first word), and orthographic context for word types. Then, we update the pre-trained model with our newly trained model and apply the resulting model to the analyst report dataset. Given that each sentence should contain at least three words, the dataset consists of 660,166 sentences.

The transformation of the texts into a TDM and the resulting reduction and weighting of words are often considered pre-processing steps, but in this paper, it takes place within the model. This decision is sensible because the parameter variation for constructing the TDM plays a central role in the dictionary induction.

4.2. Stock market data

The dictionary induction depends on stock prices and events, i.e., earnings releases, to measure abnormal returns and derive word polarities. Refinitiv Datastream provides data on the earnings releases, which start in January 2009 and last until the end of 2019. The event study further requires one year of stock price data before the first earnings release date, as mandated by the estimation window length. Accordingly, stock prices from 2008 until the end of 2019 are used for the companies in our dataset. As a reference market, the S&P 500 index is utilised. The total return index is used, which incorporates adjustments for stock splits and dividends. The total number of events amounts to 1,272 earnings releases.

5. Word polarisation

5.1. Event study setup

The analyst reports' texts are related to the abnormal returns of the companies analysed in the reports. This approach is sensible, as both the capital market and financial analysts react to the publication of new information. In the research setting, these are the earnings releases of the companies. Huang et al. (2014) find this exact relationship for analyst reports in their study. EventStudyTools (Schimmer et al., 2014) provides an implementation to carry out the event study using the market model. Following Henry (2008) and Skinner and Sloan (2002), three trading days provide the first guidance for the event window. The window is centred on the day of the earnings release. Accounting literature recommends using a short event window to reduce confounding events and suggests considering the day before the events. Many companies release negative news before earnings releases due to tactical reasons (Henry, 2008; Skinner & Sloan, 2002). Analyst reports released up to ten days after an earnings release are included to consider both analyst reports published immediately after an earnings release and reports published after a few days. The estimation window is set to 250 trading days (McWilliams & Siegel, 1997; Thompson, 1995).

In the following dictionary induction, a regression model relates the sentiments to abnormal returns. A potential further modification of the study setup would be to discretise the sentiment scores and abnormal returns to compare the resulting classifications. However, since this could lead to the loss of important information for creating word lists, this potential modification was not implemented. Another design decision is whether only events with significant abnormal returns are used. On the one hand, it could improve the signal-to-noise ratio when selecting the most relevant words. On the other hand, it reduces the list of potentially usable analyst reports, and additionally, the dictionary would have been trained using mostly extreme situations.

Figure 3 shows the average abnormal returns across all companies. The values on the x-axis are the days relative to the earnings releases. Moreover, the figure shows the average abnormal returns per day per company and the cumulative sum of the average abnormal returns per company five days before and after the earnings releases. This figure provides the first intuition for the calculated abnormal returns. It also suggests that the three-day event window found in the literature could be roughly appropriate.

Furthermore, we want to analyse in more detail which event window provides a good compromise between sufficient coverage of the time period and the clarity of the signal. To this end, Figure 4 shows the size of the event window on the x-axis. The proportion of significant tests is shown on the y-axis. It should be noted that the (inclusive) range [0,1] is used for the event window size for two days. In the other cases, symmetrical windows are used (e.g., [−1,1] corresponds to 3 days and [−3,3] corresponds to 7 days). The available data indicates that an event window of [0,1] shows the largest ratio of significant abnormal returns based on the window size. However, from another perspective, analysing all *CARs* shows that the absolute value of the average absolute *CARs* is approximately equal to the [−1,1] window. Simultaneously, the theoretical considerations

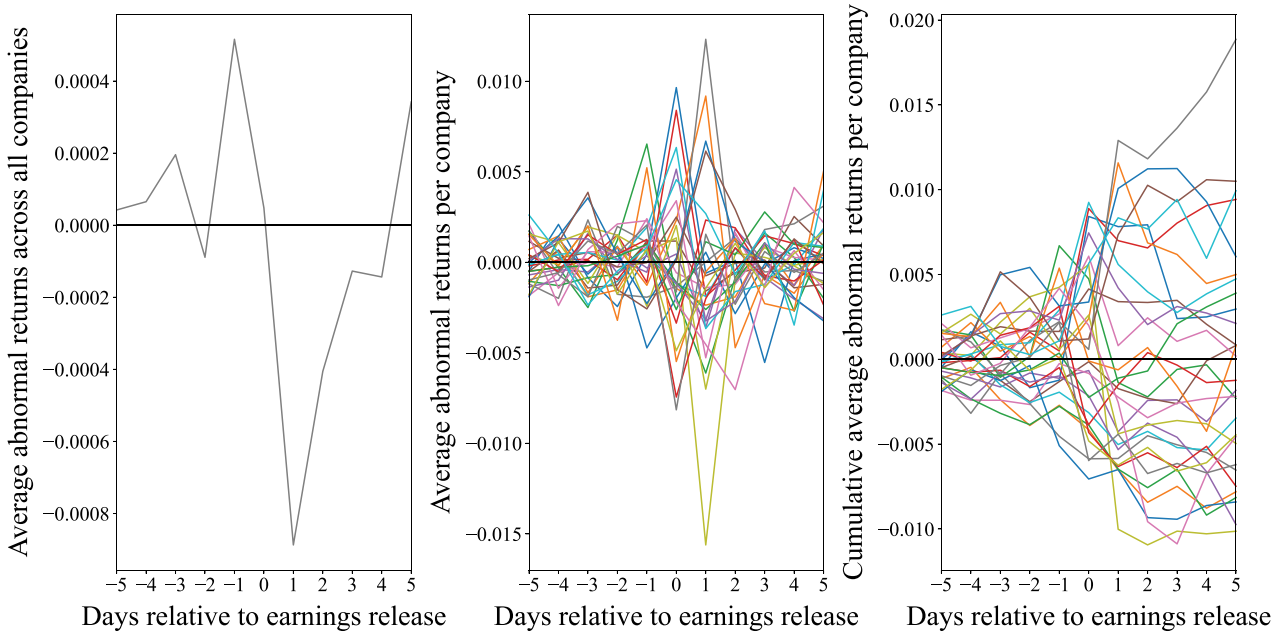

Figure 3. Overview of abnormal returns relative to earnings releases.

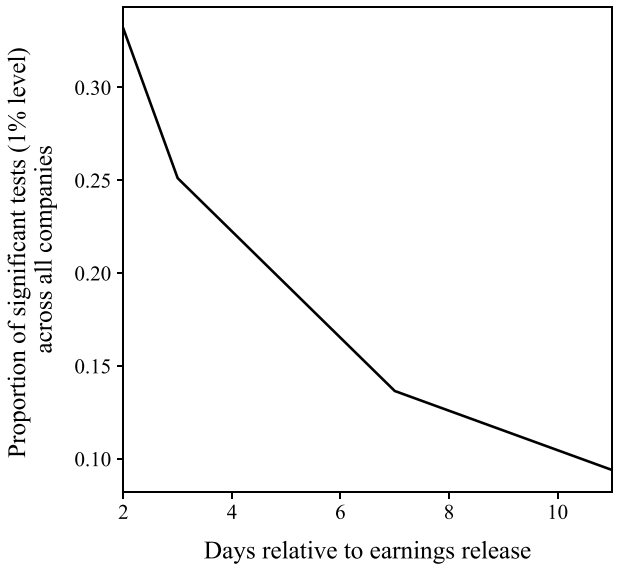

Figure 4. Identification of the appropriate event window size.

mentioned above must be kept in mind, as companies may publish information one day before the event. For this reason, [−1,1] is chosen as an event window for this study.

Figure 5 shows a heatmap to provide further insights into how the abnormal returns are distributed across firms and quarters. The colour of the cells reflects the abnormal returns for the associated event in percentage points. Furthermore, in cases where the *t*-test is significant at the 1-percent level, the cell annotation is displayed. The chosen kind of visualisation can help to identify systematic clustering or irregularities concerning a company or quarter. It further helps to deepen the data understanding. For the significant negative CARs, we can observe clustering on some occasions. For Cisco, multiple significant negative returns can be observed during the second half of 2010. Additionally, IBM exhibits a strong pattern of negative CARs, which are significant in many cases. The significant positive CARs show less obvious patterns and are distributed more evenly across companies and time.

5.2. Model training

The word polarisation is based on all analyst reports from 2009 to 2017. The subsequent evaluation uses the remaining reports of 2018 and 2019. That results in 11,972 reports (81.4 %, 1,033 earnings releases) for polarisation and 2,737 reports (18.6 %, 239 earnings releases) for evaluation. We ensure that some reports do not carry too much weight in the analysis. Therefore, reports of earnings releases are deleted, for which less than three reports are available. This reduces the polarisation dataset to 11,917 reports. We also winsorise the abnormal returns at the 1-percent level to avoid a severe influence of outliers on the coefficient estimate (minimum/maximum thresholds: −15.14 % and 11.21 %). Figure 6 demonstrates that this corrects the outliers which were present in the data.

Table 2 shows that there is sufficient report coverage per company in the evaluation dataset. Above all, it provides a comparison of the two datasets and indicates that there is no structural difference regarding the mean of the CARs and the corresponding standard deviations. The datasets are different but still well suited for comparison. Further, the numbers show that there are no unusual deviations from the mean values for individual companies.

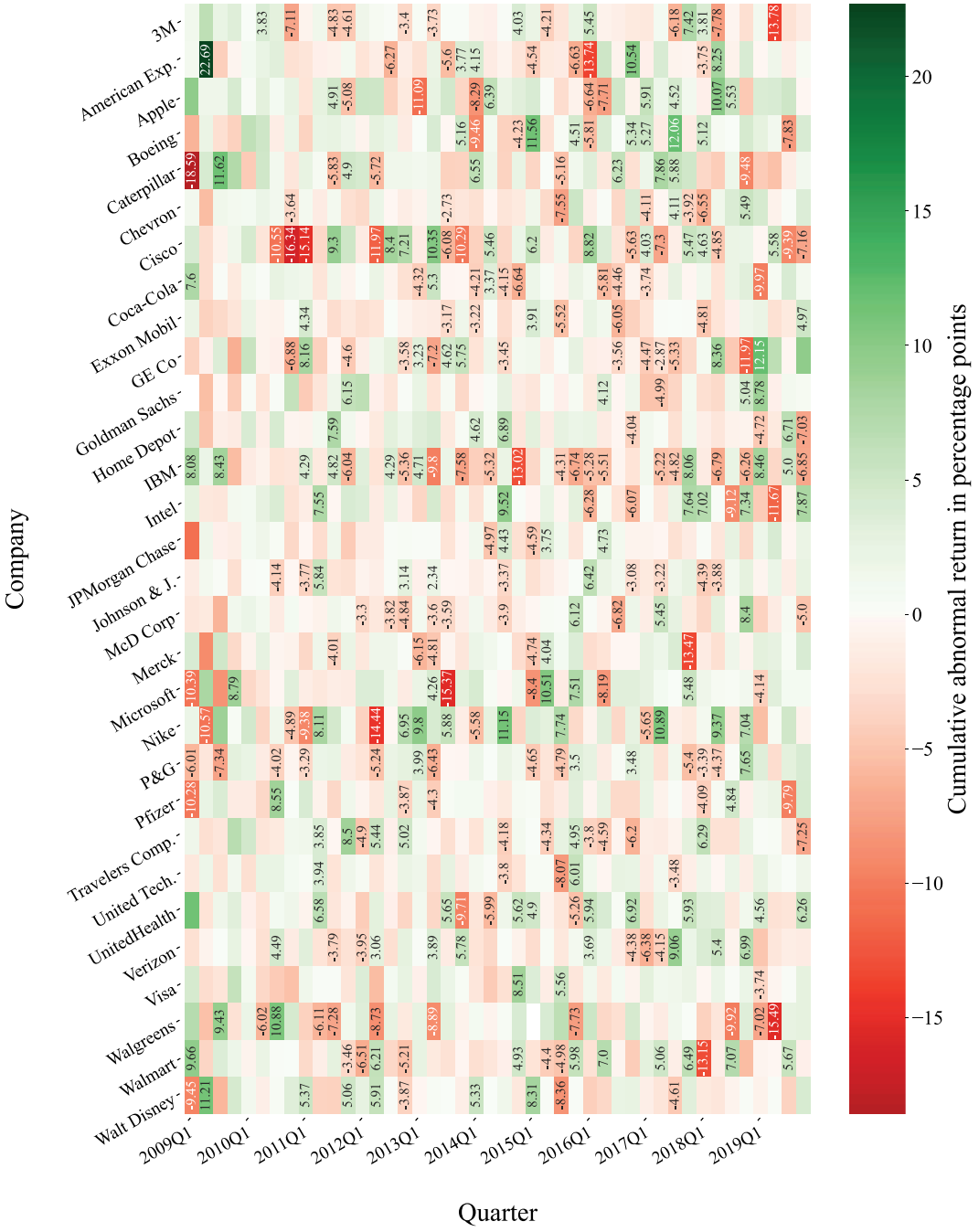

Figure 5. Heatmap of cumulative abnormal returns broken down by company and quarter. Cell annotations are shown for events that are significant at the 1-percent level.

For creating the polarity word lists, the polarisation dataset is split again. One part can be used to train the model and another part can be used to test it. Group k-fold cross-validation is used to increase the generalisability. Thereby the process of word polarisation can be repeated among differently composed data samples. The group k-fold procedure ensures that a group is not included in the training and test dataset for each fold at the same time. In this case, groups consist of companies. The k-fold splitting algorithm splits the polarisation dataset into five parts (each contains six companies). By doing so, companies and quarters are roughly equally represented during data splitting. After the specific model setup has been trained and tested for each fold, the mean performance is calculated. Figure 7 visualises the splitting of the dataset. The k-fold procedure is repeated for each setup in the induction process. For the following grid search, the parameters are varied for each setup in the procedure described hereafter. Finally, the setup that yields the best results in explaining abnormal stock price returns is chosen.

Since each fold's data is composed differently, a TDM for each fold is initialised from the analyst reports. The tf–idf transformation normalises the term counts. Loughran and McDonald (2011) recommended this for sentiment analysis. The terms are weighted according to their frequency in the entire

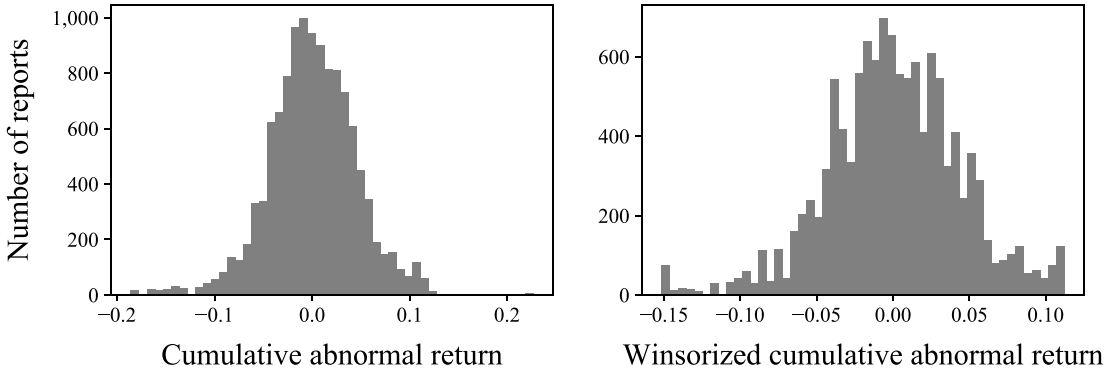

Figure 6. Cumulative abnormal returns of the polarisation dataset without (left) and with winsorising at the 1-percent level (right).

Table 2. Descriptive statistics for the 30 companies in the dataset. Each variable is split into the values for the polarisation and the evaluation dataset.

Company	Number of reports		CAR		CAR std.	
	polarisation	evaluation	polarisation	evaluation	polarisation	evaluation
3 M Co	209	50	0.0031	−0.0576	0.0335	0.0638
American Express Co	467	111	−0.0062	0.0126	0.0453	0.0386
Apple Inc	1,039	194	0.0076	0.0257	0.0420	0.0430
Boeing Co	501	105	0.0097	0.0142	0.0430	0.0391
Caterpillar Inc	305	107	0.0103	−0.0400	0.0576	0.0305
Chevron Corp	229	72	−0.0064	0.0161	0.0280	0.0371
Cisco Systems Inc	766	121	−0.0043	−0.0094	0.0680	0.0539
Coca-Cola Co	277	89	−0.0107	−0.0075	0.0296	0.0480
Exxon Mobil Corp	213	68	−0.0102	−0.0015	0.0242	0.0305
General Electric Co	306	82	−0.0114	0.0263	0.0350	0.0786
Goldman Sachs Group Inc	218	59	−0.0043	0.0150	0.0324	0.0404
Home Depot Inc	193	57	0.0109	−0.0152	0.0296	0.0381
Intel Corp	801	164	0.0078	−0.0130	0.0356	0.0722
IBM Corp	554	109	−0.0180	−0.0106	0.0481	0.0543
Johnson & Johnson	336	70	−0.0005	−0.0013	0.0279	0.0288
JPMorgan Chase & Co	307	62	−0.0090	0.0051	0.0270	0.0251
Mcdonald's Corp	316	59	0.0019	−0.0020	0.0293	0.0403
Merck & Co Inc	330	61	−0.0044	0.0087	0.0415	0.0240
Microsoft Corp	686	179	0.0053	0.0067	0.0559	0.0191
Nike Inc	483	126	0.0142	0.0160	0.0591	0.0481
Pfizer Inc	273	59	−0.0027	−0.0090	0.0260	0.0487
Procter & Gamble Co	347	88	−0.0053	0.0087	0.0317	0.0409
Travellers Companies Inc	127	53	−0.0117	−0.0053	0.0288	0.0411
United Technologies Corp	282	63	−0.0011	0.0088	0.0267	0.0125
UnitedHealth Group Inc	253	66	0.0049	0.0171	0.0423	0.0322
Verizon Communications Inc	430	80	−0.0025	0.0105	0.0339	0.0384
Visa Inc	399	123	0.0053	0.0019	0.0362	0.0205
Walgreens Boots Alliance	386	76	0.0028	−0.0265	0.0557	0.0656
Walmart Inc	329	83	−0.0003	−0.0161	0.0391	0.0675
Walt Disney Co	555	101	0.0026	−0.0023	0.0438	0.0238
Sum	11,917	2,737				
Mean			0.0004	0.0007	0.0440	0.0480

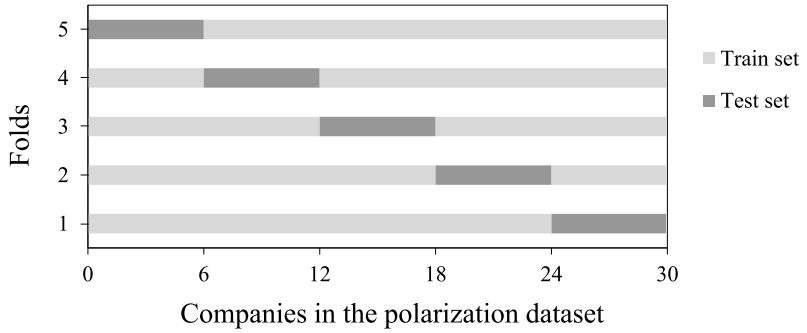

Figure 7. Groupwise k-fold splitting for dictionary model training and testing.

dataset. Words that frequently occur in the dataset are weighted lower per document. Building the TDM also involves setting upper and lower limits on the percentage of documents in which a word can occur to remain in the dataset. On the one hand, it is possible to identify recurring words that do not add any value to the sentiment analysis, as they frequently occur both in positive and negative settings. On the other hand, words can be removed that occur rarely and might not be able to represent actual patterns for positive or negative sentiment.

As explained in detail in section 4.1, since commonly co-occurring words have already been identified and concatenated, it is not necessary to explicitly include all possible bigrams into the resulting TDM. The primary motivation is that the inclusion of all conceivable bigrams leads to a drastic increase in the dimensionality of the TDM. Due to the inherently limited number of events/data, this should be avoided if possible. Utilising the tf–idf matrix implies that the data is now stored in a bag of words representation. That is, the original order of the words is no longer kept. We can directly annotate each document in the TDM with the abnormal returns of the corresponding earnings releases.

Building on the preliminary steps, a regression model is used to determine which features have a particularly strong influence in explaining abnormal returns. In the setup, we follow the work of Pröllochs et al. (2015) and implement the ridge regression model. The ridge regression regularises the coefficients of the features (unigrams and bigrams). That is, all features are preserved, but the magnitude of the coefficients has decreased. This regularisation approach is especially helpful when the coefficients of a few features are exceedingly high, and the model is strongly driven by these features. Formally, the ridge regression is defined as follows (Hastie et al., 2009):

$$\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \tag{9}$$

The formula shows the two essential parts of the ridge regression model. We can see that the sum of squared residuals is complemented with a penalisation term. It controls the shrinkage based on the λ variable. If the value for λ is increased, the coefficients' magnitude decreases, as they are penalised more strongly. This type of regression allows to potentially increase the R^2 out-of-sample, as overfitting to the training data is at least partially negated by the shrinking term. Dummy variables are used for the companies to account for company-specific influences. Furthermore, dummies for each year are introduced. We also control for industries, because ideally, these should not be represented by the dictionary (Loughran & McDonald, 2011). The North American Industry Classification System (NAICS) provides the respective industry class. The same features are used for the training and test set because the model has been trained using only these features. Therefore, features that are part of the test but not of the training set are excluded from each fold's regression.

The following parameters are varied to identify a hyperparameter configuration that achieves the best possible cross-validated R^2 for the regression: the λ of the ridge regression; the maximum percentage of documents in which a feature can occur (df_{max}), and the minimum percentage of documents in which a feature can occur (df_{min}). From our point of view, values for df_{min} higher than 0.02 hardly make sense since words must otherwise occur too frequently in the dataset. Even with a value of 0.02, one word or group of words is contained in every fiftieth document. The same is true for the lower limit of df_{min}. In our opinion, a word should appear at least in every five hundredth document (0.002). The parameter is adjusted with a step size of 0.002. The parameter df_{max} is varied in the range from 0.2 to 0.9 with a step size of 0.1. Likewise, as initial tests have shown, values for λ are varied between 6 and 16. They are varied in steps of two. Altogether this leads to 480 model combinations for the grid search.

While identifying the best model in terms of test dataset performance is usually the primary goal, it is desirable to choose a less overfitted model to the training data in the case of models with similar performance. Normally, the adjusted R^2 would be calculated for this purpose. However, the adjusted R^2 is problematic if a model has more features than observations (please refer to James et al. (2013) and Fahrmeir et al. (2013) for more details on the handling of high-dimensional datasets). In our case, the discrepancy can even be very large. Additionally, if the number of features and the number of observations is about the same, the adjusted R^2 cannot be interpreted meaningfully. During our analyses, we also found that cross-validation and the regularisation of ridge regression alone are not fully capable of limiting overfitting to the extent desirable for feature reduction necessary to induce a sentiment dictionary. Despite the minimum and maximum values for word frequencies in the data set, we calculate models with over 15,000 features in our grid search. Considering that we ultimately want to develop polarity word lists that are manageable in size, it is necessary to significantly reduce the number of features at the early stages of dictionary development. Of course, this must be done while considering any loss in model performance. Therefore, we want to give the test score a higher weight than the training score. This could be done, for example, by creating a ranking for test and train score. A high test score and a low train score would

Table 3. Mean R^2 scores for ridge regression using different λ and TDM parameters.

Balanced test mean R^2	Test mean R^2	Train mean R^2	λ	*Features*	df_{min}	df_{max}	description
0.0147	0.0599	0.2444	10	3,730	0.012	0.7	highest balanced test mean R^2
0.0140	0.0533	0.2019	14	2,355	0.020	0.7	highest ranking combination
0.0122	0.0631	0.3267	6	9,180	0.004	0.7	highest test mean R^2
0.0097	0.0574	0.3390	6	15,567	0.002	0.3	highest train mean R^2
0.0051	0.0326	0.2049	16	15,487	0.002	0.2	lowest test mean R^2
0.0102	0.0437	0.1879	16	2,206	0.02	0.2	lowest train mean R^2

each be assigned a low number, and both would be added together, giving the test score a higher weighting, e.g., multiply by two. One would then choose the model with the lowest adjusted ranking score. Since we do not want to commit ourselves to a self-determined multiplier, we use a different approach and first put the mean test score of the cross-validation in relation to its training score to get a measure of overfitting. Then, we multiply the overfitting score again with the mean test score. Thus, we chose a criterion that is specifically suited for reducing a disproportion between train and test mean R^2 while keeping the relative importance of the test score:

$$balanced\ test\ mean R^2 = test\ mean R^2 * \frac{test\ mean R^2}{train\ mean R^2} \tag{10}$$

The most informative configurations of the grid search are listed in Table 3. For the best model with the highest balanced test mean R^2 (0.0147), the test mean R^2 is 0.0599. At first glance, this seems to be relatively small but is also within the range measured by Pröllochs et al. (2015) in a similar setting. The final model setup has the following parameters: λ: 10; df_{min}: 0.012; df_{max}: 0.7; *features*: 3,730. Although it is for our case desirable to arrive at a low number of features, i.e., the most important features are kept, the results indicate that this is impossible without losing substantial model performance. Table 3 demonstrates that it is important to find a good balance between model size and the level of λ in the ridge regression. Overfitting is present in a setup with a train mean R^2 of 0.3390. In this case, we can assume that the model does not generalise well. Even by adjusting the λ parameter, this cannot always be solved sufficiently. We also see that the balanced R^2 prevents us from selecting the model with the highest test R^2, which requires 9,180 features. We would have selected a model with an increased test mean R^2 of 6.3 % (0.0599 to 0.0631) while expanding the number of features by 146 % (3,730 to 9,180).

At this point, one could compare the ridge regression with other approaches (e.g., support vector machine or random forest). However, we do not pursue a broad cross-model comparison further. Our primary goal is not finding the definite best model but rather an in-depth analysis of this specific research setup to identify the most salient terms.

Table 4. Top 30 positive and negative features resulting from ridge regression for analyst reports. The numbers are the weights (coefficients) of the ridge regression.

Positive features
better expected: 0.05248; beat: 0.041915; raising: 0.037639; raising estimates: 0.034552; upside: 0.034219; raising price: 0.032567; raising eps: 0.030799; strength: 0.029719; strong: 0.029156; sdn: 0.028364; increase: 0.028302; flu: 0.027972; raised: 0.027713; better feared: 0.027096; improved: 0.026765; raise: 0.026067; order growth: 0.024239; much better: 0.023215; improvement: 0.022488; improving: 0.021898; stabilizing: 0.021806; across: 0.021597; better: 0.021559; stronger: 0.019887; rises: 0.019745; eps beat: 0.019574; signings: 0.019531; raising target: 0.019001; positive: 0.018396; strong results: 0.017841
Negative features
miss: −0.066819; weakness: −0.065529; disappointing: −0.053651; lowering: −0.040499; shortfall: −0.039386; lower: −0.036491; issues: −0.036255; search: −0.035773; disappointed: −0.034955; lowered: −0.031911; weak: −0.031601; lowering eps: −0.028949; headwinds: −0.028381; pressure: −0.02817; lowering price: −0.027518; lowering estimates: −0.026533; weaker: −0.026029; weaker expected: −0.025243; cut: −0.024551; near term: −0.024434; disappointment: −0.023702; missed: −0.023086; cycles: −0.022293; reset: −0.022092; impacted: −0.021383; guidance: −0.021349; box: −0.020981; enterprise spending: −0.019778; cutting: −0.018846; due: −0.018676;

5.3. Word list creation

The features of the trained model are now composed of unigrams and bigrams, which were identified separately. The fitted ridge regression model assigns a coefficient to each feature, represented by a unique term from the analyst reports. The coefficients are sorted in ascending and descending order. Table 4 shows the top 30 features with the highest positive and lowest negative influence in the model. Accordingly, these words can be interpreted as strong indicators of positive or negative sentiment, as the large coefficients were assigned despite the ridge regression's penalisation. These are almost exclusively polarity words.

Loughran and McDonald (2011) note that managers can use word lists with negative polarity words to adapt their texts to their advantage. For this reason, they consider it useful to create relatively exhaustive word lists for their dictionary. It is conceivable that financial analysts may also edit their texts accordingly,

but we see a much weaker incentive. Following Henry (2008), this word list's goal is to be as comprehensible as possible while being easy to interpret. To provide a middle ground for the FA dictionary, the LM dictionary's polarity words are added if they are part of the ridge regression features.

The ridge regression helps to identify possible polarity words automatically. However, a manual review of these is still necessary to ensure that the FA dictionary consists of words that can generally be regarded as analyst-specific polarity words. It is sensible to evaluate how the dataset can be narrowed down beforehand to limit the manual effort. Therefore, the following analysis aims to determine which limits must be set to select positive and negative model features. The features are separated by the ridge regression's positive and negative coefficients and sorted by the model's parameter values. The positive list contains 1,898, and the negative list 1,832 features. The feature lists are divided into 10-percent quantiles. The number of top positive and negative features is varied and a dictionary for each combination is created. The respective dictionaries are used to calculate the sentiment scores per analyst report. The number of negative polarity words is subtracted from the number of positive polarity words to determine the score, as shown in equation (11). The two basic variants are i) dividing by the number of polar words, i.e., $n_{positive} + n_{negative}$, or alternatively ii) dividing by the number of total words. As this approach considers the total number of words, we chose the latter, which is especially relevant for lengthy analyst reports. For example, otherwise, a 5000-word report that contains two positive and one negative term would be assigned a sentiment polarity of 0.334, which would significantly overstate the prominence of the polar terms. Hence, according to variant ii), the resulting value is divided by the total number of words contained in the text to adjust for the text length:

$$sentiment\ polarity = \frac{n_{positive} - n_{negative}}{n_{total}} \tag{11}$$

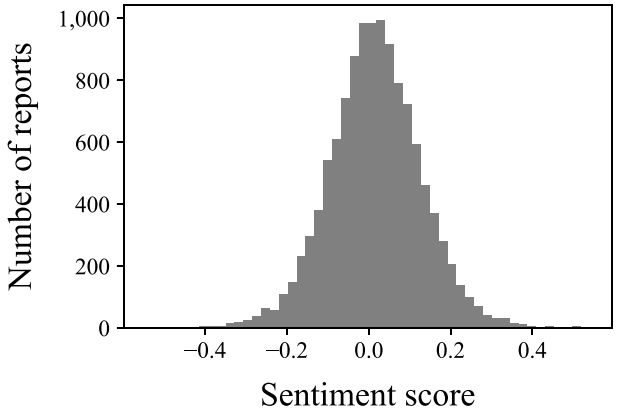

Figure 8. Histogram of the sentiment scores for the polarisation dataset.

A regression analysis relates the sentiment polarity scores to the previously calculated corresponding abnormal returns to determine the dictionary's quality. The entire polarisation dataset is used for this purpose. The distribution of sentiment scores (Figure 8) indicates an approximate normal distribution of the dataset, which seems sensible given the context.

Figure 9 shows the various cut-offs and the dictionaries' associated quality in terms of explaining abnormal returns. The word lists are pruned as far as possible. We keep the highest R^2 of 0.162, i.e., the top 40 % of the positive list (759 features) and the top 30 % of the negative list (549 features). All features that can be considered inappropriate for representing sentiment are removed manually by two researchers. For example, the following words/phrases are discarded from the positive word list: systems, announcements, importantly, gross margin, and services. And from the negative word list, for example, these words are discarded: around, expectations, watch, enough, and actually. The removal reduces the length of the positive list to 185 and the negative list to 145 features.

5.4. Combining dictionaries

The LM lists are now compared with the 3,730 textual features of the ridge regression and pruned accordingly. For words of the LM lists that do not occur in the analyst reports, it is assumed they are not vital to analyst communication. Next, words labelled positive by the proposed approach but classified negative in the LM dictionary are removed from the LM list. The remaining words of the LM lists are added to the polarity lists. The deletion of duplicates follows this. They occur if the words are already included in the manually checked lists (42 positive and 45 negative words). With this procedure, it is ensured that all LM words that were not included in the manual analysis but are part of the regression features are included in the FA dictionary (21 positive and 37 negative words). This results in relatively compact polarity lists of 206 positive and 182 negative words/phrases (see Table 5 and Table 6).

5.5. Feature weighting

The basic FA dictionary consists of positive and negative terms. However, a possible extension is to add weighting (hereafter referred to as the *wFA* dictionary). For this purpose, the ridge regression model is trained again. In this case, however, the features consist exclusively of the terms contained in the FA dictionary. The same hyperparameters are used as before. The procedure finally allows considering the individual weighting of the terms. While the weights are also provided, it is ultimately for the user to decide whether

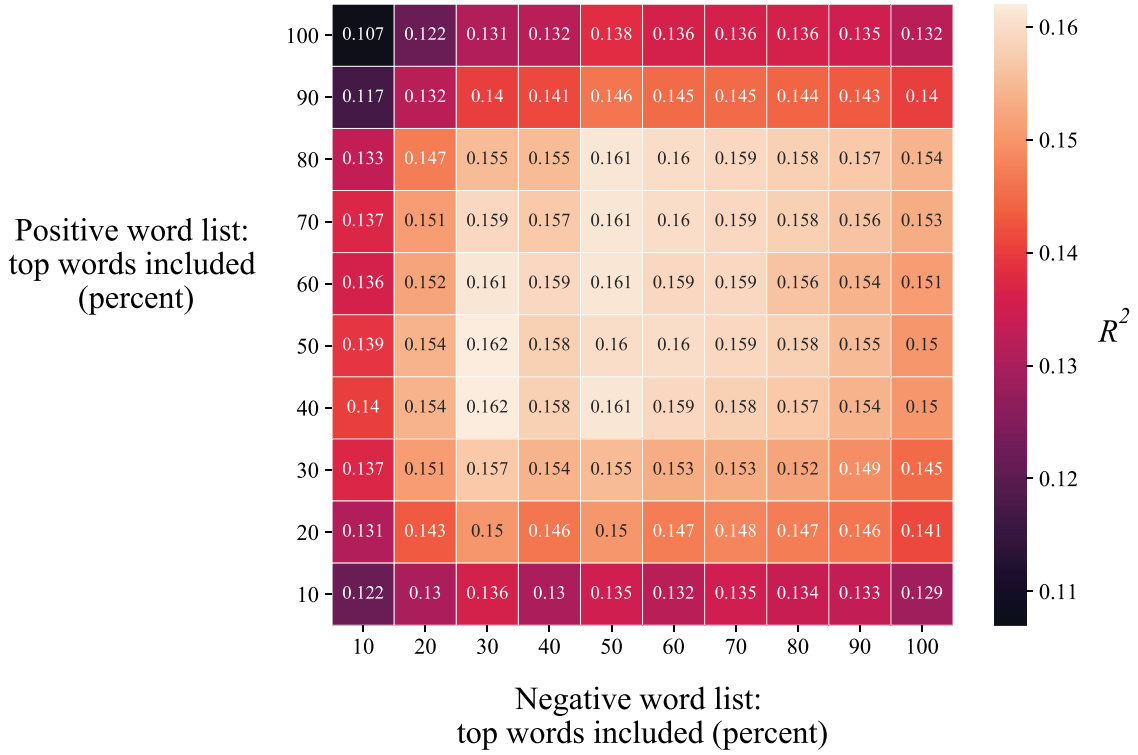

Figure 9. R^2 for explaining abnormal returns with sentiment scores for different combinations of the number of words contained in the polarity word lists.

to use them. The results in Table 7 indicate that using the optional weights can further improve the FA dictionary's performance, and it performs better than the other dictionaries. Furthermore, the FA dictionary can keep up with the deep-learning model FinBERT even without weighting. For the approach with weighted terms, it can be seen that the wFA dictionary is well suited for explaining the variance in *CARs* even better than the FinBERT model. On the one hand, it must be considered that our dictionaries were trained on the dataset we used for the comparison with the other dictionaries on the polarisation dataset. On the other hand, our dictionaries may perform worse than the FinBERT model after manually selecting polarity words and phrases. This implies that we have discarded many features from the dataset that are not polarity words/phrases but may have a significant role in explaining abnormal returns in the regression model. However, in the end, it turns out that our dictionaries still show a good performance.

6. Evaluation

6.1. Explanation of abnormal returns

The evaluation dataset only considers reports assigned to earnings releases from 2018 and 2019, amounting to 2,737 reports (18.6 % of the dataset). This enables a straightforward evaluation of how well the FA dictionary performs when applied to previously unknown data. The FA dictionary is compared with the polarity word lists of the General Inquirer, the Henry dictionary, the LM dictionary, and FinBERT. The sentiment scores for each dictionary are calculated based on equation (11). The FinBERT model classifies all sentences in a report (with probabilities for the categories positive, neutral, and negative), and we calculate the mean classification score afterwards. Then, a regression analysis between sentiments and abnormal stock returns is conducted. The results show that for the evaluation dataset, the FA dictionary shows a better ability to explain abnormal returns using the extracted sentiment values (see Table 8). The Henry dictionary, General Inquirer, and the LM dictionary achieve R^2s of 0.036, 0.004, and 0.023. For the induced FA dictionary, an R^2 of 0.06 is found. This value is lower than for the FinBERT (R^2 of 0.065) model, but with weighting, the wFA dictionary performs best (R^2 of 0.086). The development steps that led to these results could demonstrate how an existing dictionary in a finance context can be extended semi-automatically, thereby addressing *RQ 1*. Moreover, regarding *RQ 2*, the results show that the FA dictionary achieves superior performance in relating sentiment to abnormal returns, compared to existing dictionaries currently used in the finance domain.

Table 5. FA dictionary: positive polarity words/phrases. *Source* indicates the following: manually selected data of the analyst reports (ARe), LM dictionary (LM), and both (ARe_LM).

Positive words/phrases	Weight	Source						
better feared	0.0410	ARe	mid teens	0.0098	ARe	beat estimate	0.0056	ARe
better expected	0.0361	ARe	profitability	0.0098	ARe_LM	acceleration	0.0056	ARe
stabilizing	0.0268	ARe_LM	diverse	0.0097	ARe	profit	0.0055	ARe
comfortably	0.0267	ARe	adjusted eps	0.0096	ARe	top	0.0055	ARe
raising estimates	0.0264	ARe	nice	0.0096	ARe	enable	0.0055	ARe_LM
raising price	0.0261	ARe	improvement	0.0096	ARe_LM	easily	0.0055	LM
rises	0.0254	ARe	helping	0.0095	ARe	revenue grew	0.0054	ARe
raising	0.0242	ARe	rich	0.0094	ARe	reiterate overweight	0.0054	ARe
much better	0.0236	ARe	best	0.0094	ARe_LM	reiterate buy	0.0054	ARe
beat	0.0231	ARe	beat consensus	0.0093	ARe	solid performance	0.0054	ARe
increasing price	0.0212	ARe	relief	0.0093	ARe	benefit	0.0053	ARe_LM
raising eps	0.0212	ARe	capitalize	0.0093	ARe	rise	0.0052	ARe
outpaced	0.0204	ARe	exceeded consensus	0.0092	ARe	advantages	0.0052	LM
outperforming	0.0192	ARe_LM	recovery	0.0092	ARe	greatest	0.0051	LM
boosted	0.0189	ARe_LM	beat driven	0.0092	ARe	tailwind	0.0050	ARe
raising target	0.0189	ARe	service revenue	0.0092	ARe	ahead consensus	0.0050	ARe
encouraging	0.0188	ARe_LM	outperformed	0.0091	ARe_LM	benefiting	0.0049	ARe_LM
cost cutting	0.0179	ARe	faster expected	0.0090	ARe	better	0.0047	ARe_LM
recurring revenue	0.0178	ARe	positive impact	0.0089	ARe	boost	0.0046	ARe_LM
driven strength	0.0177	ARe	diversification	0.0089	ARe	grow	0.0046	ARe
rebounded	0.0175	ARe_LM	beat raise	0.0089	ARe	expansion	0.0044	ARe
high margin	0.0174	ARe	benefited	0.0086	ARe_LM	organic sales	0.0044	ARe
order growth	0.0172	ARe	strong	0.0086	ARe_LM	bull case	0.0044	ARe
solid execution	0.0166	ARe	raise eps	0.0085	ARe	higher	0.0044	ARe
favorably	0.0164	ARe_LM	stabilization	0.0084	LM	good	0.0043	LM
driven better	0.0162	ARe	exceeded expectations	0.0083	ARe	high end	0.0042	ARe
raise	0.0160	ARe	ahead expectations	0.0083	ARe	upbeat	0.0042	ARe
stronger expected	0.0159	ARe	collaboration	0.0083	LM	share gains	0.0042	ARe
well ahead	0.0159	ARe	sustainability	0.0082	ARe	innovations	0.0040	LM
upward	0.0156	ARe	potential upside	0.0082	ARe	positive	0.0040	ARe_LM
strength	0.0151	ARe_LM	improved sequentially	0.0081	ARe	ahead estimate	0.0040	ARe
moving forward	0.0148	ARe	increased driven	0.0081	ARe	ahead forecast	0.0039	ARe
better anticipated	0.0146	ARe	increasing estimates	0.0080	ARe	rev growth	0.0038	ARe
likely continue	0.0145	ARe	substantially	0.0080	ARe	performed well	0.0037	ARe
raised	0.0142	ARe	company raised	0.0078	ARe	slightly better	0.0037	ARe
rebounding	0.0142	LM	resilient	0.0078	ARe	buyback	0.0036	ARe
success	0.0134	ARe_LM	share buyback	0.0077	ARe	exceeded estimate	0.0035	ARe
benefitting	0.0132	ARe_LM	enabled	0.0077	LM	momentum	0.0035	ARe
sustain	0.0130	ARe	nicely	0.0077	ARe	tailwinds	0.0035	ARe
double digits	0.0126	ARe	rewards	0.0076	ARe_LM	solid	0.0035	ARe
positively	0.0126	ARe_LM	came better	0.0075	ARe	cost controls	0.0033	ARe
stronger	0.0124	ARe_LM	beating	0.0074	ARe	solid results	0.0032	ARe
increasing eps	0.0123	ARe	outperformance	0.0071	ARe	orders grew	0.0031	ARe
upside	0.0122	ARe	came well	0.0071	ARe	opportunities	0.0028	LM
funding	0.0121	ARe	helped	0.0070	ARe	upside driven	0.0026	ARe
earnings beat	0.0121	ARe	innovative	0.0068	ARe_LM	cost reductions	0.0025	ARe
rally	0.0121	ARe	margin expansion	0.0068	ARe	growing	0.0025	ARe
benefitted	0.0120	ARe_LM	accretive	0.0067	ARe	strengthen	0.0024	ARe_LM
dividend increase	0.0119	ARe	margin improvement	0.0067	ARe	outperform	0.0023	ARe_LM
sustainable	0.0119	ARe	improvements	0.0066	ARe_LM	acquisitions	0.0023	ARe
eps beat	0.0117	ARe	stability	0.0064	ARe_LM	accelerates	0.0021	ARe
take advantage	0.0117	ARe	increasing	0.0064	ARe	buy rating	0.0020	ARe
promising	0.0115	ARe	beat expectations	0.0063	ARe	stable	0.0019	LM
competitive	0.0115	ARe	comfortable	0.0063	ARe	driven higher	0.0018	ARe
smart	0.0114	ARe	strong performance	0.0062	ARe	popular	0.0015	LM
impressed	0.0113	ARe_LM	meaningful	0.0062	ARe	leadership	0.0014	LM
impressive	0.0112	ARe_LM	enhancements	0.0062	ARe_LM	accelerating	0.0012	ARe
improving	0.0111	ARe_LM	balanced	0.0061	ARe	favorable	0.0010	ARe_LM
highlight	0.0111	ARe	win	0.0060	ARe_LM	ahead street	0.0010	ARe
low cost	0.0110	ARe	enhancing	0.0060	LM	well	0.0009	ARe
improved	0.0110	ARe_LM	profitable	0.0060	ARe_LM	efficient	0.0007	LM
strong results	0.0110	ARe	management raised	0.0060	ARe	great	0.0005	LM
benefits	0.0109	ARe	easier	0.0060	LM	progresses	0.0004	LM
increase	0.0107	ARe	sequential improvement	0.0060	ARe	highest	0.0004	LM
upgraded	0.0106	ARe	progress	0.0059	ARe_LM	confident	0.0004	LM
lift	0.0106	ARe	encouraged	0.0059	ARe_LM	improves	0.0003	ARe_LM
able	0.0104	LM	volume growth	0.0059	ARe	reported solid	0.0002	ARe
driven strong	0.0101	ARe	ahead	0.0058	ARe	revenue growth	0.0001	ARe
cost reduction	0.0100	ARe	optimism	0.0057	ARe			

Table 6. FA dictionary: negative polarity words/phrases. *Source* indicates the following: manually selected data of the analyst reports (ARe), LM dictionary (LM), and both (ARe_LM).

Negative words/phrases	Weight	Source						
miss	0.0452	ARe_LM	negatively impact	0.0126	ARe	difficult	0.0068	ARe_LM
weakness	0.0431	ARe_LM	share loss	0.0126	ARe	reducing	0.0065	ARe
disappointing	0.0423	ARe_LM	lower end	0.0125	ARe	worse	0.0062	ARe_LM
disappointed	0.0420	ARe_LM	hurt	0.0125	ARe_LM	defensive	0.0061	LM
shortfall	0.0372	ARe_LM	missing	0.0125	ARe	loss	0.0059	ARe_LM
lowering price	0.0363	ARe	argue	0.0121	LM	missed estimate	0.0059	ARe
lowering estimates	0.0340	ARe	bad	0.0121	ARe_LM	tough	0.0057	ARe
disappointment	0.0325	ARe_LM	long	0.0120	ARe	driven lower	0.0055	ARe
enterprise spending	0.0301	ARe	commitments	0.0119	ARe	outside	0.0055	ARe
lowering	0.0297	ARe	expiration	0.0119	ARe	faces	0.0054	ARe
issues	0.0270	ARe	underperformance	0.0115	ARe_LM	decelerating	0.0053	ARe
reset	0.0238	ARe	inflation	0.0115	ARe	lost	0.0052	LM
lowering eps	0.0234	ARe	weak	0.0114	ARe_LM	slowing	0.0052	ARe_LM
lowered eps	0.0233	ARe	reducing eps	0.0112	ARe	owing	0.0051	ARe
pullback	0.0223	ARe	fell short	0.0111	ARe	slower growth	0.0051	ARe
declined due	0.0221	ARe	rebound	0.0110	ARe	negatively	0.0050	LM
weaker expected	0.0220	ARe	holiday season	0.0110	ARe	erosion	0.0050	LM
lowered	0.0217	ARe	unearned revenue	0.0110	ARe	breakdown	0.0049	LM
cutting	0.0215	ARe	deferred revenue	0.0108	ARe	sluggish	0.0049	LM
missed consensus	0.0213	ARe	diluted share	0.0107	ARe	lose	0.0049	LM
costs associated	0.0213	ARe	concern	0.0106	ARe_LM	late	0.0048	LM
slowed	0.0205	ARe_LM	challenges	0.0106	ARe_LM	cautious	0.0047	ARe
severe	0.0203	ARe_LM	pressures	0.0105	ARe	volatile	0.0047	LM
higher tax	0.0194	ARe	demand creation	0.0105	ARe	fees	0.0047	ARe
negatively impacted	0.0192	ARe	pay	0.0102	ARe	unable	0.0046	LM
unfortunately	0.0186	ARe_LM	deceleration	0.0102	ARe	caution	0.0046	ARe_LM
competitive pressures	0.0183	ARe	transition	0.0101	ARe	downside	0.0044	ARe
caused	0.0177	ARe	pain	0.0101	ARe	recession	0.0043	LM
interest rates	0.0172	ARe	reduce	0.0101	ARe	adjust	0.0043	ARe
weaker	0.0171	ARe_LM	decrease	0.0101	ARe	drag	0.0042	LM
sequential growth	0.0171	ARe	problems	0.0098	ARe_LM	questions	0.0042	LM
headwinds	0.0166	ARe	spend	0.0098	ARe	deterioration	0.0041	ARe_LM
pressured	0.0166	ARe	lower expected	0.0097	ARe	slowdown	0.0041	ARe_LM
cause	0.0164	ARe	charge	0.0097	ARe	persistent	0.0040	LM
pass	0.0163	ARe	smaller	0.0097	ARe	misses	0.0039	ARe_LM
concerning	0.0163	ARe	opposed	0.0096	ARe_LM	slowly	0.0039	LM
softer	0.0160	ARe	despite	0.0096	ARe	forced	0.0039	LM
cut	0.0157	ARe_LM	soft	0.0095	ARe	exposed	0.0037	LM
pause	0.0155	ARe	poor	0.0094	ARe_LM	issue	0.0035	ARe
pressure	0.0155	ARe	divested	0.0092	LM	disclosed	0.0034	LM
missed	0.0154	ARe_LM	revision	0.0092	ARe	headwind	0.0031	ARe
softness	0.0154	ARe	revised guidance	0.0092	ARe	underperform	0.0031	LM
strike	0.0149	ARe	slow	0.0087	ARe_LM	weakest	0.0030	ARe_LM
drop	0.0149	ARe	problem	0.0086	ARe_LM	dropped	0.0030	LM
competitive environment	0.0148	ARe	deteriorating	0.0085	ARe_LM	weakened	0.0029	LM
lower	0.0147	ARe	delay	0.0085	ARe_LM	challenge	0.0028	LM
limiting	0.0146	ARe	absence	0.0084	LM	concerned	0.0028	ARe_LM
weakening	0.0145	ARe_LM	offset lower	0.0083	ARe	force	0.0028	LM
decelerated	0.0144	ARe	share losses	0.0082	ARe	question	0.0027	LM
revised eps	0.0142	ARe	behind	0.0081	ARe	worst	0.0027	LM
tougher	0.0141	ARe	trimming	0.0081	ARe	concerns	0.0024	ARe_LM
disappoint	0.0141	ARe_LM	lagging	0.0081	ARe_LM	declines	0.0024	ARe_LM
slower	0.0135	ARe_LM	delinquencies	0.0079	LM	restructuring	0.0020	LM
lack	0.0132	ARe_LM	divestiture	0.0078	LM	lackluster	0.0018	LM
high single	0.0132	ARe	due lower	0.0078	ARe	reduced	0.0015	ARe
dropping	0.0132	ARe	continued weakness	0.0078	ARe	recall	0.0009	LM
divestitures	0.0131	ARe_LM	inability	0.0078	LM	underperformed	0.0006	LM
low end	0.0131	ARe	due	0.0075	ARe	negative	0.0004	ARe_LM
depreciation	0.0131	ARe	closing	0.0075	LM	downward	0.0002	LM
trimmed	0.0130	ARe	unlikely	0.0073	ARe	decline	0.0002	ARe_LM
overhang	0.0128	ARe	regulatory	0.0069	ARe			

Table 7. Comparison between dictionaries by R^2 applied to the polarisation dataset.

Dictionary	R^2
General Inquirer	0.016
Henry	0.039
LM	0.051
FinBERT	0.098
FA	0.096
wFA	0.140

Table 8. Comparison between dictionaries by R^2 applied to the evaluation dataset.

Dictionary	R^2
General Inquirer	0.004
Henry	0.036
LM	0.023
FinBERT	0.065
FA	0.060
wFA	0.086

6.2. Matching with human-observed sentiment polarity

The first and central evaluation step of the dictionary development was to compare the R^2 of the different sentiment dictionaries. In this way, the relationship to the financial market, in particular, could be analysed. However, the question arises whether our developed FA dictionary is also comparable or better suited to capture the sentiment perceived by humans. This evaluation step can be seen as complementary to the prior one, and we understand it as a form of triangulation. It is reasonable to assume that the FA dictionary, which was trained using capital market reactions, could perform worse. After all, there may be certain words that are relevant to the capital market reaction but are not perceived by a human being in this way. To verify this, we split the analyst reports into individual sentences. Then, we select a random sentence from each earnings announcement assigned to the company and period in question. For eight draws, this results in 1,904 sentences. The sentences were then manually annotated with the following categories (number of occurrences are in parentheses): positive (819), neutral (668), negative (417). As an evaluation metric, the *F1* score with micro averaging is used to account for the class imbalance.

Table 9. Classification accuracy measured with *F1* score (micro) for human-annotated labels.

Dictionary	*F1*
General Inquirer	0.443
Henry	0.515
LM	0.478
FinBERT	0.757
FA	0.549
wFA	0.541

As shown in Table 9, both the FA and wFA dictionaries show good classification performance for the sentiment perceived and accordingly annotated by humans. Of the dictionaries, the Henry dictionary comes the closest in terms of classification accuracy. Nevertheless, its performance compared to the FA and the wFA dictionaries is 6.6 % and 5.05 % worse, respectively. Besides, the Henry dictionary scored significantly worse in explaining the abnormal returns. We also include a comparison with the FinBERT model, as it can be considered state of the art. In our view, this is particularly important because we can thereby contribute to closing the research gap between research relying mostly on machine learning models and research originating from the finance domain. The FinBERT model shows by far the best performance for pure sentiment determination. However, this is not surprising since this large and complex model has already been fine-tuned using analyst reports throughout its training. In favour of the dictionaries, of course, it must be considered that they have 1) universal applicability and 2) an easier interpretability. Depending on the respective field of application and purpose of use, our results can provide some guidance for researchers and practitioners alike.

In addition to the forecast quality, the manual annotation also allows us to verify the validity of the assigned weights in a straightforward way. For this purpose, each of the labelled documents is linked to the occurrence of the words (unigrams and bigrams) contained in the FA dictionary. This makes it possible to examine the percentage distribution of each sentiment word among the three categories. The analysis only covers words that occurred at least ten times in the annotated sentences. Table 10 shows the words with the most positive and negative weight. For the

Table 10. Comparison of assigned weight and the distribution across manually labelled sentences for the ten words with the biggest/smallest weight (at least ten occurrences in the labelled sentences were required).

Word	Count	Weight	Positive (%)	Neutral (%)	Negative (%)
raising	18	0.0242	94	0	6
beat	31	0.0231	84	13	3
raise	13	0.0160	62	31	8
strength	35	0.0151	86	14	0
raised	16	0.0142	88	6	6
stronger	11	0.0124	73	18	9
upside	25	0.0122	60	16	24
competitive	23	0.0115	30	22	48
improved	20	0.0110	80	10	10
benefits	19	0.0109	47	32	21
weakness	11	−0.0431	9	18	73
issues	12	−0.0270	25	33	42
weaker	11	−0.0171	18	9	73
headwinds	15	−0.0166	20	20	60
pressure	24	−0.0155	13	25	63
lower	65	−0.0147	28	23	49
long	33	−0.0120	45	33	21
spend	15	−0.0098	53	20	27
despite	33	−0.0096	45	24	30
due	52	−0.0075	35	21	44

positive words, we see a relatively strong agreement between the assigned weights and the annotations (columns "positive", "neutral", and "negative"). The word "competitive" seems to be the strongest outlier since the assignment is less distinct, compared to "raising" for example. In the case of negative words, there is also a fairly strong agreement. We have to be careful to draw deeper conclusions from Table 10 because 1) many of the words with a high weight did not occur with sufficient volume in this limited sample, and 2) the limit of 10 occurrences is too low to make statistically valid statements. Nevertheless, we think that this analysis can further help to understand our results from an additional perspective.

7. Discussion

Preceding research indicates a lack of scientific literature that develops and evaluates domain-specific sentiment dictionaries (Nassirtoussi et al., 2014). In particular, identifying and developing effective methods offers the potential to advance research and practical applications. Our extension of the widely used domain-specific LM dictionary (which was induced for a different but related domain – 10-K filings) demonstrates how a domain-specific dictionary can be created based on capital market data (arguably more objective labelling) and with considerably less manual effort. Naturally, the mapping employed in the present approach between sentiment scores and abnormal returns must be carried out differently in other contexts (application domains), e.g., sales figures, page views, or transactions. At the same time, such a specific mapping is a strength of the described approach.

We show that the FA dictionary yields better results than both domain-specific and general-purpose dictionaries. In this context, Huang et al. (2014) compare their Naïve Bayes sentiment classifier, which achieves an accuracy of 80.89 %, with other established finance-related sentiment dictionaries for their dataset, which gained 62.02 % (Loughran & McDonald, 2011) and 65.44 % (Henry, 2008). For the General Inquirer and DICTION, the paper lists 48.40 % and 54.93 % accuracy, respectively. Even if this paper does not primarily analyse a classification problem but a regression problem, these results point in the same direction as the performance differences observed in this study. Our classification results using FinBERT lead in the same direction as those given for the Naïve Bayes sentiment classifier. A comparison to DICTION is not performed because Loughran and McDonald (2015) have already shown that this dictionary is not suitable for an application in the finance domain.

Huang et al. (2014) note that negative words have a stronger influence. Negatively classified analyst reports influence share prices more than positively classified reports. Based on the conducted analysis and concerning Figure 9, the results do not suggest a structurally stronger influence of negative polarity words than positive polarity words. Furthermore, Loughran and McDonald (2016) highlight the drawbacks of testing word lists with positive words since negations can have too much of a distorting effect. We leave it to the end-users whether they want to work with the positive word list and make it available. Our implemented phrase detection can identify commonly co-occurring word combinations, and this also includes negations, e.g., "not thrilled". However, such word combinations are not included in the final lists of the FA dictionary because they do not seem to have a considerable influence in explaining abnormal stock returns. Two-syllable words are deleted in our text clean-up. Consequently, words such as "no", cannot be used for detecting negations. However, we have decided to avoid selectively favouring one word over another to circumvent a decision problem. This indicates that the treatment of negations in sentiment dictionaries is an interesting problem. We partially reflect this through phrase detection in our approach. At the same time, we think there is still a lot of potential for future research to rethink the sentiment dictionary induction process from the ground up, incorporating negation detection and phrase detection.

Loughran and McDonald (2011) support a tf–idf weighting to prevent words that frequently occur from carrying too much weight in textual analysis. Henry and Leone (2015) consider this a risk and do not recommend using any weighting scheme since the word weighting, and thus the sentiment depends on the entire dataset and its size and composition. Because the word list creation in this study is based on a large dataset, a middle ground was chosen. The tf–idf weighting is used in the first part of the suggested approach but not in the final dictionary evaluation. The rationale is that the dictionary should be applicable to singular sentences or a collection of a few sentences. In those cases, a meaningful tf–idf weighting would not be possible. Hence, an evaluation using the weighting would potentially overstate the actual accuracy, which we want to avoid.

It is worthwhile to mention that our dictionary approach puts a special emphasis on reducing the number of features as much as possible until a significant amount of information is being lost. Thus, when solely focusing on increasing test performance, we advise refraining from using the balanced test meanR^2. In particular, if there is a high discrepancy between the train and test score, this can be particularly detrimental to the overall score. For similar text mining approaches to the dictionary induction, dimensionality reduction is important, and our proposed formula addresses this issue. Balancing

model performance and data input is a tough choice when conventional machine learning approaches cannot provide the required solutions.

Twedt and Rees (2012) measure report complexity by the Fog Index and De Franco et al. (2015) show that experienced financial analysts write more readable analyst reports. Moreover, a positive correlation between the readability and the number of companies covered by an analyst is identified. Supposing the readability of analyst reports differs from the readability of other document types in the finance domain, which provided the data fundament for existing sentiment dictionaries, it can be argued that inducing an analyst-specific sentiment dictionary is especially important. Furthermore, the Hypertextual Finance Glossary (Harvey, 1999) might be suitable to examine the use of finance-specific terminology in analyst reports and compare it with other finance-related document types.

Naturally, this study is subject to limitations. First, the data sample is U.S.-centric. Moreover, if a report is written particularly negatively, but there is a particularly positive abnormal return, the sentiment dictionary's usability may be limited. In that case, positive words would be associated with negative abnormal returns or vice versa. Hence, the assumption that abnormal returns are related to sentiment underlies the dictionary induction. A disadvantage in our study setup is that the dictionaries used for comparison were not necessarily induced based on abnormal stock returns.

Regarding the data source of the analyst reports, it must be considered that all analyst reports were sourced from the Thomson Reuters Advanced Analytics platform, potentially biasing the data. Although there is a wide range of reputable companies among the included brokers, it is still conceivable that not-included brokers do not cooperate with Thomson Reuters, or there are other licencing reasons. Furthermore, the proposed data cleansing procedure may produce slightly different results for reports of other data vendors. For the large-scale implementation of the procedure, it would be interesting to determine if and to what extent the weighting of the identified terms varies based on differences between data vendors.

For practice applications, this paper provides an easy-to-apply sentiment dictionary, which conforms to the traditional structure of categories of polar words. It can be applied using standard analytic approaches and provides a superior classification performance compared to the other dictionaries. Besides, the proposed dictionary induction procedure and practical considerations provide guidance to practitioners and contribute to the knowledge base of dictionary induction at the same time. The conducted evaluation is valuable by comparing the developed FA dictionary to others and bridging the gap to recent deep neural network models. By comparing the classification performance, practitioners aiming to analyse analyst reports in an automated fashion obtain insights on the performance differences and potential compromises that need to be made.

8. Conclusion and future research

The domain-specific sentiment dictionary developed in this paper extends a domain-specific dictionary (induced for a different but related domain) in a semi-automated fashion. In particular, our approach is dedicated to the communication of financial analysts. The induced FA dictionary shows superior results in analysing sentiments of analyst reports compared to more general-purpose dictionaries. Similar to Jegadeesh and Wu (2013) and Pröllochs et al. (2015), a regularised linear model is used to relate textual content to stock returns to extract words associated with abnormal stock returns. Beyond that, our approach also recognises frequently occurring phrases of two words. Words/phrases associated with positive (negative) returns are assigned a positive (negative) sentiment label. The regularisation of a ridge regression model helps to identify particularly influential words while at the same time reducing the impact of overfitting. On this basis, we combine our word lists with the LM dictionary and create an analyst-specific dictionary. Through our approach, we are also able to provide the weights of the individual words/phrases from the previously set up regression model. As a result, the user is no longer bound to weigh words equally but can differentiate between the influence of a polarity word/phrase. In the end, our approach results in two relatively short polarity lists in which a weight is given for each word/phrase. In addition, the FA dictionary can be applied to short text passages or individual sentences without the need for word weighting in the context of a larger document collection. This is especially important for practical applications, e.g., when determining the sentiment of short news items. Thus, the standard FA dictionary and its enhanced version, the weighted FA dictionary (wFA) are immediately ready to use.

The evaluation of the FA dictionary's performance is implemented using a sample of the analyst reports not used for model training during the polarisation phase of the dictionary induction. The FA (wFA) dictionary is compared to polarity word lists from established dictionaries, including the General Inquirer (Stone et al., 1966), Henry (2008), and Loughran and McDonald (2011), but also to a deep-learning approach, i.e., the FinBERT classifier of Yang et al. (2020). In the evaluation, the proposed wFA dictionary (with a weighting of the polarity words) proves to be best suited to measure the sentiment in analyst

reports when related to abnormal stock returns. Even without weighting polarity words, the FA dictionary provides superior results to the dictionary-based approaches. However, our evaluation is not limited to explaining abnormal stock market returns but also examines how the dictionary performs in classifying human-annotated sentences. This allows us to determine whether the approach based on abnormal stock returns can be used in a broader application field. For the evaluation based on manually annotated sentences, the classification accuracy, measured by the *F1* score, is higher when compared to established dictionaries but expectedly lower compared to the FinBERT model. We thus provide a tool that can be used immediately in research as well as in practical applications for various text mining tasks.

Furthermore, we want to take a step back and integrate the evaluation results into the broader field of text analytics in finance. First of all, it is important to emphasise that the created FA dictionary should be placed in context with the existing dictionaries. Sentiment dictionaries are often used in different contexts and by different users compared to more complex and resource-intensive models such as FinBERT. A sophisticated, domain-specific machine learning model will be superior to a classic sentiment dictionary after extensive training. In contrast to this, sentiment dictionaries are particularly suitable for intuitive use and a high degree of explainability. For example, it is easy to capture the direct term counts and obtain a better understanding of the forecasts. By also providing weightings, the user can further differentiate between words/phrases and potentially perform more precise analyses. In our opinion, both types of sentiment classification approaches, dictionary- and machine-learning-based, will continue to have their rightful place in the future.

Future research could address multiple remaining questions. First, an extension of the available data would be beneficial, both in terms of data size and data heterogeneity between different brokers. The analysis of a long-time horizon could uncover further crucial terms or phrases, and it could also further underline the robustness of the FA dictionary. An extension in terms of diversity refers to the analysis and evaluation related to further countries, continents, and time periods. Furthermore, the suitability of the FA dictionary for the contributions of analysts to conference calls following the publication of quarterly results might be analysed.

Disclosure statement

No potential conflict of interest was reported by the author(s).

References

Antweiler, W., & Frank, M. Z. (2004). Is all that talk just noise? The information content of internet stock message boards. *The Journal of Finance*, *59*(3), 1259–1294. https://doi.org/10.1111/j.1540-6261.2004.00662.x

Cambria, E., Poria, S., Gelbukh, A., & Thelwall, M. (2017). Sentiment analysis is a big suitcase. *IEEE Intelligent Systems*, *32*(6), 74–80. https://doi.org/10.1109/MIS.2017.4531228

Chen, X., Cheng, Q., & Lo, K. (2010). On the relationship between analyst reports and corporate disclosures: Exploring the roles of information discovery and interpretation. *Journal of Accounting and Economics*, *49*(3), 206–226. https://doi.org/10.1016/j.jacceco.2009.12.004

Das, S. R. (2014). Text and context: Language analytics in finance. *Foundations and Trends in Finance*, *8*(3), 145–261. https://doi.org/10.1561/0500000045

Das, S. R., & Chen, M. Y. (2007). Yahoo! For Amazon: Sentiment extraction from small talk on the web. *Management Science*, *53*(9), 1375–1388. https://doi.org/10.1287/mnsc.1070.0704

De Franco, G., Hope, O. K., Vyas, D., & Zhou, Y. (2015). Analyst report readability. *Contemporary Accounting Research*, *32*(1), 76–104. https://doi.org/10.1111/1911-3846.12062

Demers, E. A., & Vega, C. (2014). Understanding the role of managerial optimism and uncertainty in the price formation process: Evidence from the textual content of earnings announcements.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2019)*, Minneapolis, Minnesota. http://dx.doi.org/10.18653/v1/N19-1423 .

Fahrmeir, L., Kneib, T., Lang, S., & Marx, B. (2013). *Regression: Models, methods and applications* (1. Ed. ed.). Springer.

Hart, R. P. (2000). *Diction 5.0. Retrieved 11/ 25/2020from* http://rhetorica.net/diction.htm

Harvey, C. R. (1999). *Campbell R.* Harvey's hypertextual finance glossary. Retrieved November/ 25/2020 from http://people.duke.edu/~charvey/Classes/wpg/glossary.htm

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2. Ed. ed.). Springer.

Henry, E. (2008). Are investors influenced by how earnings press releases are written? *The Journal of Business Communication*, *45*(4), 363–407. https://doi.org/10.1177/0021943608319388

Henry, E., & Leone, A. J. (2015). Measuring qualitative information in capital markets research: Comparison of alternative methodologies to measure disclosure tone. *The Accounting Review*, *91*(1), 153–178. https://doi.org/10.2308/accr-51161

Hu, M., & Liu, B. (2004). Mining and summarizing customer reviews. *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Seattle, WA:Association for Computing Machinery.

Huang, A. H., Lehavy, R., Zang, A. Y., & Zheng, R. (2017). Analyst information discovery and interpretation roles:

A topic modeling approach. *Management Science*, *64*(6), 1–23. https://doi.org/10.1287/mnsc.2017.2751

Huang, A. H., Zang, A. Y., & Zheng, R. (2014). Evidence on the information content of text in analyst reports. *The Accounting Review*, *89*(6), 2151–2180. https://doi.org/10.2308/accr-50833

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (1. Ed. ed.). Springer.

Jegadeesh, N., & Wu, D. (2013). Word power: A new approach for content analysis. *Journal of Financial Economics*, *110*(3), 712–729. https://doi.org/10.1016/j.jfineco.2013.08.018

Kearney, C., & Liu, S. (2014). Textual sentiment in finance: A survey of methods and models. *International Review of Financial Analysis*, *33*, 171–185. https://doi.org/10.1016/j.irfa.2014.02.006

Li, F. (2010a). The information content of forward-looking statements in corporate filings—a naïve bayesian machine learning approach. *Journal of Accounting Research*, *48*(5), 1049–1102. https://doi.org/10.1111/j.1475-679X.2010.00382.x

Li, F. (2010b). Textual analysis of corporate disclosures: A survey of the literature. *Journal of Accounting Literature*, *29*, 143–165.

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–167. https://doi.org/10.2200/S00416ED1V01Y201204HLT016

Loughran, T., & McDonald, B. (2011). When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. *The Journal of Finance*, *66*(1), 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x

Loughran, T., & McDonald, B. (2015). The use of word lists in textual analysis. *Journal of Behavioral Finance*, *16*(1), 1–11. https://doi.org/10.1080/15427560.2015.1000335

Loughran, T., & McDonald, B. (2016). Textual analysis in accounting and finance: A survey. *Journal of Accounting Research*, *54*(4), 1187–1230. https://doi.org/10.1111/1475-679X.12123

Lu, B., Tan, C., Cardie, C., & Tsou, B. K. (2011). Joint bilingual sentiment classification with unlabeled parallel corpora. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, OR:Association for Computational Linguistics.

MacKinlay, A. C. (1997). Event studies in economics and finance. *Journal of Economic Literature*, *35*(1), 13–39. https://www.jstor.org/stable/2729691

Mahyoub, F. H., Siddiqui, M. A., & Dahab, M. Y. (2014). Building an arabic sentiment lexicon using semi-supervised learning. *Journal of King Saud University – Computer and Information Sciences*, *26*(4), 417–424. https://doi.org/10.1016/j.jksuci.2014.06.003

McWilliams, A., & Siegel, D. (1997). Event studies in management research: Theoretical and empirical issues. *Academy of Management Journal*, *40*(3), 626–657. https://doi.org/10.5465/257056

Mengelkamp, A. (2017). Informationen zur Bonitätsprüfung auf Basis von Daten aus sozialen Medien, Cullivier Verlag: Goettingen, Germany. Association for Information Systems.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Drean, J. (2013). Distributed representations of words and phrases and their compositionality. *Proceedings of the 26th Conference of Advances in Neural Information Processing Systems*. Lake Tahoe, USA.

Nassirtoussi, A. K., Aghabozorgi, S., Wah, T. Y., & Ngo, D. C. L. (2014). Text mining for market prediction: A systematic review. *Expert Systems with Applications*, *41*(16), 7653–7670. https://doi.org/10.1016/j.eswa.2014.06.009

Oliveira, N., Cortez, P., & Areal, N. (2017). The impact of microblogging data for stock market prediction: Using twitter to predict returns, volatility, trading volume and survey sentiment indices. *Expert Systems with Applications*, *73*, 125–144. https://doi.org/10.1016/j.eswa.2016.12.036

Peng, H., Cambria, E., & Hussain, A. (2017). A review of sentiment analysis research in chinese language. *Cognitive Computation*, *9*(4), 423–435. https://doi.org/10.1007/s12559-017-9470-8

Pröllochs, N., Feuerriegel, S., & Neumann, D. (2015). Generating domain-specific dictionaries using bayesian learning. *Proceedings of the 23rd European Conference on Information Systems*. Münster, Germany, Association for Information Systems.

Řehůřek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA: Valletta, Malta.

Remus, R., Quasthoff, U., & Heyer, G. (2010). Sentiws – A publicly available german-language resource for sentiment analysis. *Proceedings of the 7th International Conference on Language Resources and Evaluation*. ELRA: Valletta, Malta.

Rogers, J. L., Van Buskirk, A., & Zechman, S. L. (2011). Disclosure tone and shareholder litigation. *The Accounting Review*, *86*(6), 2155–2183. https://doi.org/10.2308/accr-10137

Schimmer, A., Levchenko, A., & S., M., (2014). *Eventstudytools*. Retrieved November/ 25/2020 from http://www.eventstudytools.com

Skinner, D. J., & Sloan, R. G. (2002). Earnings surprises, growth expectations, and stock returns or don't let an earnings torpedo sink your portfolio. *Review of Accounting Studies*, *7(2-3)*, 289–312. https://doi.org/10.1023/A:1020294523516

Soltes, E. (2014). Private interaction between firm management and sell-side analysts. *Journal of Accounting Research*, *52*(1), 245–272. https://doi.org/10.1111/1475-679X.12037

Stone, P. J., Dunphy, D. C., Smith, M. S., & Ogilvia, D. M. (1966). The general inquirer: A computer approach to content analysis. *American Sociological Review*, *32*(5), 859–860.

Thompson, R. (1995). Empirical methods of event studies in corporate finance. *Handbooks in Operations Research and Management Science*, *9*, 963–992. https://doi.org/10.1016/S0927-0507(05)80073-8

Twedt, B., & Rees, L. (2012). Reading between the lines: An empirical examination of qualitative attributes of financial analysts' reports. *Journal of Accounting and Public Policy*, *31*(1), 1–21. https://doi.org/10.1016/j.jaccpubpol.2011.10.010

Xing, F. Z., Cambria, E., & Welsch, R. E. (2018). Natural language based financial forecasting: A survey. *Artificial Intelligence Review*, *50*(1), 49–73. https://doi.org/10.1007/s10462-017-9588-9

Yang, Y., UY,, M. C. S.,, & Huang, A. (2020). Finbert: A pretrained language model for financial communications. *arXiv Preprint*. arXiv:2006.08097.