Machine Learning in U.S. Bank Merger Prediction: A Text-Based Approach

Katsafados, Apostolos G. and Leledakis, George N. and Pyrgiotakis, Emmanouil G. and Androutsopoulos, Ion and Fergadiotis, Manos

Athens University of Economics and Business, Athens University of Economics and Business, University of Essex, Athens University of Economics and Business, Athens University of Economics and Business

12 June 2021

Machine learning in U.S. bank merger prediction: A text-based approach

by

Apostolos G. Katsafados[1], George N. Leledakis[1*], Emmanouil G. Pyrgiotakis[2], Ion Androutsopoulos[3], Manos Fergadiotis[3]

[1] Department of Accounting and Finance, School of Business, Athens University of Economics and Business, Greece
[2] Essex Business School, University of Essex, U.K.
[3] Department of Informatics, School of Information Sciences and Technology, Athens University of Economics and Business, Greece

Abstract

This paper investigates the role of textual information in a U.S. bank merger prediction task. Our intuition behind this approach is that text could reduce bank opacity and allow us to understand better the strategic options of banking firms. We retrieve textual information from bank annual reports using a sample of 9,207 U.S. bank-year observations during the period 1994-2016. To predict bidders and targets, we use textual information along with financial variables as inputs to several machine learning models. Our key findings suggest that: (1) when textual information is used as a single type of input, the predictive accuracy of our models is similar, or even better, compared to the models using only financial variables as inputs, and (2) when we jointly use textual information and financial variables as inputs, the predictive accuracy of our models is substantially improved compared to models using a single type of input. Therefore, our findings highlight the importance of textual information in a bank merger prediction task.

JEL classification: C63, G14, G21, G34, G40
Keywords: *Bank merger prediction; Textual analysis; Natural language processing; Machine learning*

This version: May, 2021

[*]Corresponding author: Department of Accounting and Finance, School of Business, Athens University of Economics and Business, 76 Patission Str., 104 34, Athens, Greece; Tel.: +30 210 8203459. E-mail addresses: katsafados@aueb.gr (A. Katsafados), gleledak@aueb.gr (G. Leledakis), e.pyrgiotakis@essex.ac.uk (E. Pyrgiotakis), ion@aueb.gr (I. Androutsopoulos), fergadiotis@aueb.gr (M. Fergadiotis). We would like to thank Ilias Chalkidis, Prodromos Malakasiotis, and Thanos Verousis for their valuable comments and suggestions. Apostolos Katsafados acknowledges financial support co-financed by Greece and the European Union (European Social Fund- ESF) through the Operational Programme «Human Resources Development, Education and Lifelong Learning» in the context of the project "Strengthening Human Resources Research Potential via Doctorate Research" (MIS-5000432), implemented by the State Scholarships Foundation (IKY). George Leledakis greatly acknowledges financial support received from the Research Center of the Athens University of Economics and Business (EP-2256-01). All remaining errors and omissions are our own.

1. Introduction

Over the last decades, the U.S. banking industry has experienced a severe wave of consolidation through mergers and acquisitions (M&A). Aligned with this trend, the academic literature has given increased attention to the topic of bank M&As. The vast majority of the literature focuses on investigating the shareholder wealth effects around the announcement of bank mergers (Houston et al., 2001; DeLong and DeYoung, 2007; Filson and Olfati, 2014; Leledakis and Pyrgiotakis, 2021), while other studies analyze the merger-related performance changes (Cornett and Tehranian, 1992; Cornett et al., 2006), or the efficiency effects (Rhoades 1993; 1998).

Another strand of the literature attempts to identify the characteristics of merging U.S. banks, especially from the perspective of the target (Prasad and Melnyk, 1991; Wheelock and Wilson, 2000). These studies report that smaller, less profitable, and poorly-managed banks are more attractive acquisition targets. In this respect, Katsafados et al. (2021) find that banks with more positive (negative) tone in their annual reports have a higher probability of becoming bidders (targets). However, the latter study focuses on the determinants of merger likelihood under an econometric framework. Up to date, therefore, there is a gap in the literature regarding the development of classification models in a U.S. bank merger prediction task.

In the non-financial sector, there is a plethora of studies that utilize classification models to predict M&As (Palepu, 1986; Comment and Schwert, 1995; Espahbodi and Espahbodi, 2003; Edmans et al., 2012; Routledge et al., 2017). One possible explanation on why there is no substantial empirical work on this issue for U.S. banks could be that the banking industry is inherently more opaque than other industries (Flannery et al., 2004; Blau et al., 2017). Opacity means that banking assets are hard-to-value due to their financial nature which distinguishes banks from non-bank firms (Morgan, 2002). In other words, banks hold very few physically-

fixed assets compared to other types of firms. Instead, banks primarily hold loans, which are privately negotiated transactions with their borrowers. The opaqueness of these types of assets limits the ability of investors to properly evaluate the financial condition of a bank (Huizinga and Laeven, 2012; Jones et al., 2013). Researchers in merger prediction for non-financial firms use accounting measures to evaluate the financial condition of the firm. Potential bidders are perceived to be in sound financial position, whereas potential targets may face financial constraints (Espahbodi and Espahbodi, 2003). Taken altogether, it is likely that bank opacity could be one possible reason for the lack of empirical work on bank merger prediction.

Bank opacity is inversely related to disclosure of information, as the level of bank opacity decreases with the quality of disclosure (Flannery et al., 2013; Jiang et al., 2016; Zheng 2020). Banks disclose information to the public mainly through their financial statements and annual reports. On the one hand, financial statements may not effectively reduce opacity, as banks manage their statements to smooth their earnings and circumvent the capital requirements (Ahmed et al., 1999; Beatty et al., 2002; Bushman and Williams, 2012; Gandhi et al., 2019). On the other hand, bank annual reports contain one other important source of information besides balance sheet data: textual information.

There is a growing literature on how textual information can reduce firms' valuation uncertainty and the asymmetry of information on the initial public offerings (IPOs) in the U.S. (Hanley and Hoberg, 2010; Loughran and McDonald, 2013; Jegadeesh and Wu, 2013). Collectively, these studies find that the textual information of the IPO prospectuses can mitigate the uncertain valuation of IPO firms, a fact which leads to a more accurate pricing of the newly issued shares. In a similar manner, Gandhi et al., (2019) use the sentiment of banks' annual reports to gauge financial distress. The authors argue that text is more informative than simple accounting measures, as the latter source of information could be influenced by bank managers.

This happens because over-optimism in annual reports by managers increases litigation risk (Rogers et al., 2011; Loughran and McDonald, 2013). Building on these arguments, it is reasonable to assume that the use of textual information could improve our ability to evaluate the financial condition of banks by reducing bank opacity. Hence, if this assumption is valid, textual information may also enable us to more accurately identify bidders and targets in the U.S. banking industry.

Apart from reducing bank opacity, textual information could have an additional benefit in a merger prediction task. In most cases, the choice to engage in a merger is a strategic decision for the bank, especially on the part of the bidder (Ramaswamy, 1997). Potential bidders have different characteristics from potential targets, as they are usually larger and more profitable (Becher, 2009). However, the fact that a bank is financially healthier (according to its financial statements), does not necessarily imply that its strategy is to engage in M&As. For this reason, annual reports may be more insightful regarding the bank's strategic options, as managers disclose information regarding the future prospects of their bank in these reports.

Therefore, the primary aim of this paper is to investigate whether the use of textual information from bank annual reports is meaningful in a merger prediction task. More precisely, we develop classification models to identify bidders and targets in the U.S. banking industry, and we use both textual information and financial variables as inputs in these models.

To address our research question, we collect annual reports from banks that filed the reports over the period 1994-2016. By doing so, we obtain a large sample of 9,207 U.S. bank-year observations, which includes bidders, targets, and banks that were not involved in a merger. For each year and for each bank, we retrieve the annual reports from the SEC's Electronic Data Gathering, Analysis, and Retrieval (EDGAR) website. For the purpose of our analysis, we extract textual information by creating textual features from these reports using the bag of words

approach. In fact, we use the following textual features as inputs: term frequency (TF) features, and term frequency-inverse document frequency (TF-IDF) features corresponding to words (or combination of words and bigrams). Finally, we use these textual features along with financial variables in the classification machine learning models. More precisely, we use the following models: (1) support vector machine, (2) logistic regression, (3) random forest, and (4) multilayer perceptron.

A key innovation of our study is that apart from the classical aforementioned textual features, we create textual features based on word embeddings. In addition to the frequently-used generic word embeddings, we also create our own finance word embeddings. Textual features based on word embeddings are used as inputs to the multilayer perceptron model.

As the first step in our empirical analysis, we run our models using only financial variables as inputs. In the bidder prediction task, the highest accuracy score is 63.6%, while in the target prediction task, the highest score is 78.8%. As a second step, we repeat our analysis by using only textual features as inputs. In general, results for bidders are improved, while results for targets are comparable (or slightly better) to those reported in the first step. These findings suggest the importance of textual information in a bank merger prediction task. As a third step, we re-run our models using jointly textual features and financial variables. Overall, the combination of textual features with financial variables substantially improves the predictive ability of our models in both tasks. To illustrate this improvement, we report the accuracy scores of the best performing models in each task. In the bidder prediction task, the multilayer perceptron models with finance word embeddings achieve an accuracy score of 72%. In the target prediction task, the best performing model is the random forest, with an accuracy score of 89.7%. To further validate our findings, we use a second evaluation measure, the area under curve, which is computed from the receiver operating characteristic curves. The inferences from this analysis

are in line with the ones obtained from the accuracy scores.

We conduct a series of robustness tests. First, we employ the bootstrap resampling method of Berg-Kirkpatrick et al. (2012) to validate the performance of our models. More precisely, in both tasks, we compare the best performing models with themselves using different types of inputs. Second, we re-run our analysis by excluding some special years from our sample. Third, we examine whether the dimensionality of our textual features has an impact on our results. Fourth, we compute the importance score for each variable in our random forest models using the Gini impurity technique of Kurt et al., (2008). By doing so, we illustrate how meaningful textual information is in our merger prediction task. Collectively, the results of these tests support our baseline findings.

Our findings could benefit all key parties of a bank merger transaction. From the regulators' perspective, identifying future acquirers may be more beneficial than identifying future targets. When acquirers grow large through M&As, they can become too-big-to-fail and enjoy oligopolistic market power (O'hara and Shaw, 1990; Demirgüç-Kunt and Huizinga, 2013). Therefore, the development of an accurate classification model could enable regulatory authorities to *a priori* evaluate any merger-related anticompetitive effects and ensure the stability of the banking industry. From the investors' perspective, identifying future acquisition targets is a profitable strategy, due to the premium paid by the acquiring bank (Brook et al., 1998). Finally, the development of an accurate classification model could also be of use to bank managers. Managers of banks who want to expand via M&As can use such a tool to identify potential targets. At the same time, financially constrained banks that have to be acquired may use such classification models to identify and attract potential bidders (Pasiouras et al., 2010).

We contribute to the literature in four main aspects. First, instead of focusing merely on predicting future acquisition targets, we also attempt to predict future acquiring banks, as this

task is more important to regulators and depositors. Second, instead of using econometric techniques to perform our task, we utilize several machine learning models, which have several advantages over traditional econometric methodologies (Mai et al., 2019). Third, we create our own finance word embeddings, which appear to be the most meaningful textual inputs in the bidder prediction task. Finally, we provide evidence that textual information can effectively complement traditional financial variables in bank merger prediction. Our interpretation for this result is that textual information reduces bank opacity, since the language used by managers in the annual reports provides a clearer picture of the financial condition of the bank and its future strategic options.

The rest of the paper is organized as follows. Section 2 describes our sample collection and our textual analysis procedure. Section 3 discusses our classification models, and Section 4 reports our empirical findings. Finally, Section 5 concludes the paper.

2. Data and textual analysis

2.1. Sample selection

To construct our dataset, we follow a three-step approach. The first step is to collect bank annual reports (10-Ks, 10-K405s, 10-KSBs, and 10-KSB40s) from the SEC's Electronic Data Gathering, Analysis, and Retrieval (EDGAR) website. To do so, we use a web-crawling algorithm, which gathers the reports and excludes all amended documents. In our primary sample, we require that banks' filing dates are between 1994 and 2016. Furthermore, we exclude 97 observations from our sample because the filing contained fewer than 2,000 words (Loughran and McDonald, 2011). Further, we also exclude 2 observations from our sample, due to the fact that 2 banks had more than one filing in the same fiscal year (we include only the first filing). By applying these criteria, our initial sample consists of 18,031 bank-year observations.

The second step is to gather bank-specific characteristics from the Federal Reserve Bank of Chicago (FRBC), as in Katsafados et al. (2021).[1] More precisely, we collect banks RSSD IDs using the Federal Reserve Bank of New York's CRSP-FRB link. Next, we use the bank names and locations (state and/or city) to merge our initial sample from EDGAR with FRBC data. By doing so, we are able to link the banks' RSSD IDs with their corresponding Central Index Keys (CIK). To ensure the maximum number of observations, we manually match banks' RSSD IDs with their CIKs using the National Information Centre (NIC) database. This matching process leaves us with a final sample of 9,207 bank-year observations consisting of 1,160 unique banks.

As a third step, we obtain our bank merger sample from the Thomson ONE database. We focus on deals announced between February, 1994 and December, 2017.[2] To filter our sample, we use the following criteria similar to Leledakis and Pyrgiotakis (2021) and Leledakis et al. (2021):

1. Both bidders and targets are commercial banks with a three-digit primary SIC code of 602, savings institutions with a three-digit primary SIC code of 603, or bank holding companies with a four-digit primary SIC code of 6712.
2. The bidder is publicly-traded. The target can be a public firm, a private firm, or an unlisted subsidiary of a publicly-traded firm.
3. All public firms are listed on NYSE, AMEX, or NASDAQ.
4. The bidder acquired an interest of more than 50% of the target firm after the merger. Before the merger, its interest was below 50%.

The above selection process results in a sample of 966 bank M&As. As described in the filter

[1] We acquire financial information of bank holding companies (BHCs) from the FR Y-9C reports and of commercial banks and savings institutions from Call Reports.

[2] To be included in our merger sample, a bank should be a bidder or a target in a twelve-month period after the filing date (Routledge et al., 2017). The earliest filing date of our sample is in the end of January, 1994 and the latest is in the end of December, 2016.

criteria, all bidders are publicly-traded. However, in the case of targets, 389 are publicly-traded, and the remaining ones are either private-owned banks or subsidiaries of listed banks. Since the sample also includes unlisted targets, our sample selection process ensures that our subsample of bidding banks includes all listed banks that had acquired another bank during our examination period.[3] Hence, from the final sample of 9,207 bank-year observations, 7,874 refer to banks that are not involved in M&As (non-involved hereafter). Table 1 reports the number of bidders, targets, and non-involved banks on an annual basis over our examination period.

Insert **Table 1** here

2.2. Financial variables

We choose to use a set of financial variables as inputs in our predictive models that satisfy the following two criteria: (i) they are likely to influence acquisition decisions (Wheelock and Wilson, 2000; Pasiouras et al., 2010), and (ii) these variables are limited in number to avoid overfitting of our models (Palepu, 1986). In what follows, we briefly describe the nine financial variables used in this study.[4]

The first two financial variables relate to the inefficient management hypothesis. According to this hypothesis, the motive behind M&As is to replace the inefficient management of the target firm (Manne, 1965). Hence, following Pasiouras et al. (2010), we employ two bank efficiency measures: the cost to income ratio (*Cost efficiency*), and the return on total assets (*ROA*). Further, we also account for the impact of size. Wheelock and Wilson (2000) find that smaller banks are more likely to become acquisition targets. Therefore, in line with Baele et al. (2015), we use the logarithm of total assets as a measure of bank size (*Size*). Capital strength is also an important determinant of bank acquisition behavior, as weaker-capitalized banks are more likely to be

[3] If we had restricted our sample to public-to-public deals, then bidding banks would be included in the non-involved subsample, a fact which could lead to biased estimates.
[4] All financial variables are measured at the most recent fiscal year end prior to the filing date.

acquired (Hannan and Rhoades, 1987; Pasiouras et al., 2007). For this reason, we use the ratio of common equity to total assets (*Capital strength*). Moreover, we control for the impact of loan activity on bank acquisition likelihood using the ratio of loans to total assets (*Loans*), as in Pasiouras et al. (2010).

Market power is a commonly-stated motive behind bank M&As (Hankir et al., 2011). Hence, we also use in our models the ratio of each bank's deposits to the total deposits of the U.S. banking industry at a given year (*Market power*). Further, acquired banks tend to have higher amounts of loan loss reserves relative to non-acquired banks (Wheelock and Wilson, 2000; Pasiouras et al., 2010). In line with these results, we include the ratio of loan loss provisions to total loans (*Asset quality*). Further, we proxy for the banks' dependence on off-balance sheet activities using the ratio of non-interest income to total income (*Non-interest income*), as in Ellul and Yerramilli (2013). Finally, in the spirit of Cornett et al. (2006), we use the ratio of deposits to total assets (*Deposits*) as a measure of liquidity. Table A1 in the Appendices provides a detailed list of the variables, along with the corresponding codes from the FR Y-9C reports for bank holding companies and the Call reports for commercial banks and savings institutions. Table 2 reports the summary statistics of all financial variables. Particularly, we split the sample into the following four categories: bidders (Panel A), targets (Panel B), non-involved (Panel C), and all (Panel D).

Insert **Table 2** here

2.3. Textual analysis and parsing methodology

2.3.1. Textual sources

All bank annual reports are encoded in the hypertext markup language (HTML). Hence, as in most studies using textual analysis in finance, we follow the parsing process of Loughran and McDonald (2011). Through this process, we remove HTML formatting and any other non-textual

information, such as embedded images or spreadsheets that might be present in the text (Bodnaruk et al., 2015). Moreover, we exclude all identified HTML tables, if their numeric character content is higher than 10%, as effectively documented by Loughran and McDonald (2014).

2.3.2. Pre-processing and bag of words

After the parsing procedure, we have to transform the textual information into numerical features before we insert them as inputs to our models. To do so, we follow the pre-processing procedure, which consists of several steps (Jegadeesh and Wu, 2013; Loughran and McDonald, 2014; Nassirtousi et al., 2014).

First, we eliminate single letter words, abbreviations, numbers, punctuation marks, and stop words (Gandhi et al., 2019). Second, we impose a minimum occurrence threshold in order to remove words with low frequency. Following Mai et al. (2019), we consider the 20,000 most frequent words of the bank annual reports of the remaining text. Third, we use the bag of words (BOW) approach to transform our unstructured textual information into inputs with explicit numerical structure. More precisely, we use the Natural Language Toolkit (NLTK) to tokenize text into individual words. As a matter of fact, this approach treats each unique word as a separate textual feature, and constructs a document-term matrix, where each row and column represent a document and a word, respectively (Loughran and McDonald, 2011).

In the textual analysis literature, raw counts of textual features are not considered the best measure of a text's information content. Therefore, we represent each textual feature using the two most widely-used term weighting schemes: (1) the term frequency (TF) normalized by document length, and (2) the term frequency-inverse document frequency (TF-IDF). *TF* is calculated as the proportion of each textual feature in each document, so it assigns an equal weight for each feature. *TF-IDF* adjusts the TF scores by putting a lower weight on features that

appear more frequently in our sample of bank annual reports (Jegadeesh and Wu, 2013; Loughran and McDonald, 2016; Katsafados et al., 2021). Prior studies suggest that *TF-IDF* is a more effective weighting scheme compared to *TF*, as it assigns lighter weights to common words, which have a less meaningful impact on textual analysis tasks (Balakrishnan et al., 2010; Brown and Tucker, 2011; Loughran and McDonald, 2011; Loughran and McDonald, 2016; Mai et al, 2019). We calculate the *TF-IDF* weight of word *i* in the j^{th} document as reported in the equation below:

$$TF\text{-}IDF(t_{ij}) = TF(t_{ij}) \times \left[-\log\left(\frac{n_i}{N} \right) \right] \quad (1)$$

where $TF(t_{ij})$ is the number of times a term *i* appears in a document *j,* divided by the total word count of the same document for normalization purposes, *N* represents the number of documents in our entire dataset, and n_i the total number of documents including at least one occurrence of the i^{th} word.

At this point, it is worth mentioning that one limitation of the BOW is that it does not control for the presence of polysemous words (words with multiple meanings) in the text. To control for this issue, we also use bigrams in our textual analysis. Bigrams are essentially word pairs, obtained using the word n-gram features (n equal to 2). The use of bigrams may improve the ability of our models to disambiguate the meaning of a polysemous word. Note that the BOW approach is also based on word n-gram features, when n equals to 1 (unigrams).

2.3.3. Word embeddings

The aforementioned BOW approach has a prevalent role in studies that employ textual analysis in finance. As mentioned before, a main drawback of this approach is that it does not account for polysemous words, an issue which can be partially resolved with the use of bigrams.

However, another drawback of the BOW approach is that it is not able to capture well the morpho-syntactic and semantic properties of the words of the text (Manning and Schutze, 1999; Kearney and Liu, 2014; Loughran and McDonald, 2016). This happens because conventional BOW models rely on the frequency of words under the assumption that each word occurs independently of all others. In this regard, it is likely that models that use conventional BOW representations as textual inputs are not fully capable of understanding the underlying semantics of the text (Loughran and McDonald, 2016). To alleviate this concern, we also employ the word embedding features to represent textual information.

The word embedding approach is a relatively new representation for textual data in natural language processing (NLP). The fundamental concept behind this model is that words with similar properties co-occur with similar neighbors (Mai et al., 2019). In other words, a word embedding is a type of word representation which allows words with similar properties to have a similar representation. More precisely, this model represents each word as a vector in a low dimensional space (Goldberg, 2017). The word embedding vector includes real values, which reflect the morpho-syntactic and semantic properties of the word.

Mikolov et al. (2013) develop the word2vec technique, where word embeddings can be produced either through the continuous bag of words (CBOW) model, or the skip-gram model. Both models use shallow neural networks to learn word representations for each unique word. The CBOW model combines the embeddings of surrounding words to predict the word in the middle of a window of text, whereas the skip-gram model tries to predict the context words in a window of text for a given word in the middle of the window.

Pennington et al. (2014) introduce an alternative method for producing word embeddings, known as global vectors for word representation (GloVe). GloVe embeddings typically lead to similar performance in NLP tasks as word2vec embeddings, but GloVe embeddings are more

readily available in different dimensionalities, and pre-trained on diverse corpora. Therefore, in our paper, we employ the available 200-dimension generic embeddings created by Pennington et al. (2014). These embeddings are obtained from 6 billion tokens from Wikipedia 2014 and Gigaword 5, and have a vocabulary size of 400K words.[5]

In our empirical setting, one possible concern with the generic word embeddings is that they are not trained on (obtained from) a finance-specific corpus. To account for this issue, we also employ domain-specific (DS) word embeddings. DS word embeddings are trained on data from a specific domain of interest. For this reason, they may be able to represent better the semantics of the text compared to generic word embeddings. In particular, we use word2vec to create our 200-dimension finance word embeddings (FWE) induced from textual disclosure in the finance domain.[6] In particular, our finance word embeddings are derived from 4.9 billion tokens of EDGAR financial disclosures from 1994 to 2016 (including all 10-K, 10-Q, and S-1 filings), and have a vocabulary size of 2.3M words.

In more detail, we employ the skip-gram model to produce our finance word embeddings. As noted earlier, the skip-gram model learns word vector representations aiming to predict the context (surrounding words in a window) from the central word of each (sliding) text window (Mikolov et al., 2013). In this regard, if we have a corpus of T words $w_1, w_2, .., w_T$, skip-gram aims to maximize the following log-likelihood objective:

$$\sum_{t=1+m}^{T-m} \sum_{-m \le i \le +m, i \ne 0} \log P(w_{t+i} \mid w_t) \qquad (2)$$

where w_t is the central word of the (sliding) window at location t in the corpus, w_{t+i} is the context

[5] These word embeddings have been proved to be efficient to many tasks. Also, they are publicly available https://nlp.stanford.edu/projects/glove/

[6] To do so, we use the free available Python library of gensim (https://radimrehurek.com/gensim/).

word at location $t+i$, and m defines the window size $(2 \times m - 1)$ of the window around w_t.[7]

Each word has two embeddings (vectors of real numbers), an input (w^{in}) and an output (w^{out}) one, which are randomly initialized, and learned by minimizing the objective. For every token w_t at position t of the corpus and every position $t+i$ (i≠0) within a window [t-m, t+m] around position t, we aim to be capable of predicting which vocabulary word occurs at position $t+i$ by multiplying (dot product) w_t^{in} and w_{t+i}^{out}. The basic form of skip-gram employs the softmax function to calculate the likelihood of a surrounding word w_{t+i} given a center word w_t:

$$P(w_{t+i} \mid w_t) = \text{softmax}\left(\mathrm{w}_{t+i}^{out} \times w_t^{in}\right) = \frac{\exp\left(\mathrm{w}_{t+i}^{out} \times w_t^{in}\right)}{\sum_{w \in V} \exp\left(\mathrm{w}^{out} \times w_t^{in}\right)} \qquad (3)$$

where V is the vocabulary. We learn the w_t^{in} and w_{t+i}^{out} by maximizing the probability we assign to the word w_{t+i} that actually occurs at each position t+i of each window. In fact, we obtain the word embeddings as follows:

$$\left\langle E^{in},\ E^{out} \right\rangle = \underset{\left\langle E^{in},\ E^{out} \right\rangle}{\arg\max} \sum_{t=1+m}^{T-m} \sum_{-m \le i \le +m, i \ne 0} \log P(w_{t+i} \mid w_t) \qquad (4)$$

where E^{in} and E^{out} are matrices that include in their columns all the in (w_t^{in}) and out (w_{t+i}^{out}) vectors of all words in the vocabulary. We maximize the objective by stochastic gradient ascent. However, in practice the softmax of $P(w_{t+i}|w_t)$ is computationally expensive, because of the large size of the vocabulary V. We, therefore, use the negative sampling version of the skip-gram model. Instead of predicting the context word w_{t+i} from the central word w_t, we now aim to be able to identify the true context word w_{t+i}, when given the true context word w_{t+i} and a randomly sampled word r (multiple randomly sampled words are used in practice, instead of just

[7] Our FWE are created with window size equal to 5.

one). In effect, instead of aiming to produce a probability distribution over the vocabulary V for position $t+i$, we now have a binary classification problem, where we need to classify w_{t+i} in the true (positive) class, and r to the false (negative) class. The objective now becomes:

$$\left\langle E^{in},\ E^{out} \right\rangle = \underset{\left\langle E^{in},\ E^{out} \right\rangle}{\arg\max} \sum_{t=1+m}^{T-m} \sum_{-m \le i \le +m, i \ne 0} \log \sigma\left(\mathrm{w}_{t+i}^{out} \times w_t^{in} \right) + \log\left[1 - \sigma\left(r^{out} \times w_t^{in} \right) \right] \quad (5)$$

where σ is the sigmoid (logistic) function, and $\sigma\left(w_{t+i}^{out} \times w_t^{in}\right)$ is the probability estimate that word w is the true context word. After maximizing the objective, we keep the vectors of E^{in} as word embeddings, though the vectors of E^{out} can also be used alternatively.

Figure 1 visualizes the position of various words from our financial word embeddings in a 2-dimensional vector space. Given that the FWE have 200 dimensions, we project them into 2 dimensions using the t-Distributed Stochastic Neighbor Embedding (t-SNE) dimensionality reduction technique. As shown in the figure, words with similar properties are located in close proximity to each other in the word embedding space. For instance, in the bottom of the figure, there is a set of words that express negativity, such as crisis, distress, weak, recession, and turmoil among others. Furthermore, words close to the upper right corner of the figure relate to merger events, such as, acquired, acquire, target, purchase, merger, and acquisition. This finding is in line with our conjecture that annual reports contain information regarding the banks' strategic choices, and particularly their M&As strategies. Finally, words close to the upper side deal with profitability issues, such as sales, revenues, increase, decrease, earning, and profit. Considering the previous facts, we can infer that our FWE serve their purpose of being specialized in financial texts.

Insert **Figure 1** here

3. Methodology

In this section, we describe the three parts of our methodological approach. First, we describe how we match the merging banks with the non-involved banks (not involved in mergers) to conduct our classification task and how we split our datasets into training set and out-of-sample (testing) set. Second, we analyze the machine learning models we use. Third, we describe the two measures we use to evaluate the performance of our models.

3.1. Matching and splitting datasets

To address our research question, we have to specify two binary models that are capable of distinguishing between: (1) bidders and non-involved and (2) targets and non-involved. To do so, we have to construct our two datasets in a proper way. The first dataset will include only bidding banks and non-involved banks, and the second dataset will include only target banks and non-involved banks. Obviously, the number of bidders and/or targets is disproportionally smaller compared to non-involved banks, which suggests that both datasets are imbalanced.

Imbalanced datasets are a common issue in classification tasks in finance, such as acquisitions or bankruptcy forecasting (Barnes, 1998, 1999; Laitinen and Kankaanpaa, 1999; Neophytou and Mar Molinero, 2004; Pasiouras et al., 2007, 2010). Following these studies, we mitigate this issue by adopting the undersampling approach of Veganzones and Severin (2018). This method generates a balanced subsample from our original sample by excluding observations from the majority category (in this case, the non-involved banks). By doing so, our first dataset consists of 966 bidders and a matched equal number of non-involved banks, and our second dataset consists of 389 targets and a matched equal number of non-involved banks. The benefit of this approach is that a balanced sample may provide more relevant information than an imbalanced sample (Imbens, 1992). We use the filing year of the banks' annual reports as the matching criterion. This matching criterion has two main benefits: (1) it helps us control for any time effects in our

analysis, (2) it allows us to include all the other variables as inputs in our models (Hasbrouck, 1985).[8]

After balancing our two datasets, we split them into training and out-of-sample datasets. Following Geng et al. (2015), Doumpos et al. (2017), and Routledge et al. (2017), we select 80% of each dataset as the training set, and the remaining 20% as the out-of-sample. The out-of-sample is selected from a future period, as the usefulness of a classification model is evaluated according to its ability to correctly predict observations that occur in the future (Espahbodi and Espahbodi, 2003).

3.2. Machine learning models

To perform our merger classification task, we use our machine learning models.[9] The machine learning models we use are: (1) support vector machine (SVM), (2) logistic regression (LOGIT), (3) random forest (RF), and (4) multilayer perceptron (MLP). SVM, LOGIT, RF, and MLP use as textual inputs the features obtained by the BOW approach. In the MLP, we further use the textual features obtained by word embeddings (generic or finance). Figure 2 illustrates this process step by step.

We note that using centroids of word embeddings is still, in effect, a bag of words approach, since word order is discarded. More powerful deep learning models, like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) can be applied to text (Goldberg, 2017), using word embeddings as inputs, in ways that consider word order. Also, more recent deep learning models for text, mostly Transformer-based models (Vaswani et al., 2017) can be pre-trained on gigantic corpora (Wikipedia and book collections) of unlabelled documents and

[8] Size is also frequently-used as a matching criterion. However, if we use size to match our datasets, then we have to exclude it from our classification models. In line with previous studies, we prefer to use size as a control variable rather than as a matching criterion, because it is an important factor in explaining merger behavior (Espahbodi and Espahbodi, 2003; Pasiouras et al., 2007, 2010).

[9] In all our models, all financial variables are standardized. Textual features are also standardized when they are combined with financial variables.

then fine-tuned (further trained) on much fewer (compared to a gigantic corpus) task-specific labelled training instances, achieving better performance than when using only the task-specific labelled training instances. However, models of this kind can so far cope only with very short documents. For example, the commonly used BERT models (Devlin et al., 2018), which employ Transformers, can typically process up to 512 sub-word tokens (sub-word tokenizers break words into smaller units). Even very recently proposed variants of Transformer-based models for "long" text (Zaheer et al., 2020) can only process text input of up to 4,096 sub-word tokens, whereas the documents we consider are much longer. By contrast, centroids of word embeddings have no input length limitation.

Insert **Figure 2** here

3.2.1. Machine learning models with bag of words approach

3.2.1.1. Support vector machine

Support vector machine (SVM) is a non-probabilistic supervised learning algorithm, first introduced by Vanek (1998). So far, several studies have used SVMs in finance tasks, such as bankruptcy forecasting (Min and Lee, 2005; Shin et al., 2005; Wu et al., 2007), stock price forecasting (Cao, 2003; Pai and Lin, 2005). Given a set of training instances that explicitly belong to various pre-defined categories, the SVM learns a decision boundary that defines the predicted identity of each instance. This decision boundary is practically a hyperplane in the feature space. The aim is to find the optimal hyperplane that maximizes the width of the gap (margin) among the instances of different categories (Kumar and Ravi, 2016). Notably, only the training samples near the hyperplane, either at the boundaries of the margin or inside the margin in case of letting "slack" in the separation, matter when creating the hyperplane. It is worth mentioning that finding the maximum margin hyperplane belongs to the general quadratic programming optimization problems. Interestingly, SVM has the advantage that is able to handle

non-linearly separable data. In such a case, it can employ non-linear kernel functions such as radial basis function (RBF) kernel. As a result, our training data are projected into a higher dimensional space so that our data become more separable. (Nassirtoussi et al., 2014). Hence, in our paper we repeat our empirical analysis using: (i) a linear SVM, and (ii) an SVM with RBF kernel.[10]

3.2.1.2. Logistic regression

Logistic regression (LOGIT) is also one of the most commonly-used models in merger prediction task (Hasbrouck, 1985; Palepu, 1986; Ambrose and Megginson, 1992; Comment and Schwert, 1995; Barnes, 1998, 1999; Powel, 2001; Espahbodi and Espahbodi, 2003; Cremers et al., 2009; Routledge et al., 2017). LOGIT is capable of handling binary classification tasks by estimating a non-linear sigmoid function between inputs and the binary output. LOGIT's rationale is to maximize the conditional log-likelihood of training samples in order to learn the parameters of the model. In fact, it typically uses stochastic gradient ascent or variants. To deal with overfitting the training dataset, regularization terms could be added to the log-likelihood. In our empirical setting, we employ L2 regularization, which subtracts the squared L2 norm of the weights vector (multiplied by a hyper-parameter), from the log-likelihood.

3.2.1.3. Random forest

Random forest (RF) is an ensemble machine learning algorithm, initially designed by Breiman (2001) as a variant of Bagging (Breiman, 1996). We employ RF by creating several uncorrelated decision tree classifiers. These decision trees are typically trained on bootstrap copies of original samples by randomly selecting a subset of features (Mai et al., 2019). The prediction process is then performed with each individual tree predicting a class. Based on majority voting, the class

[10] The hyper-parameters of our SVM models are tuned based on the 5-fold cross-validation performance of the training set.

with the most votes becomes the output of our model. In general, RF outperforms the classical decision trees (DT), since it addresses the DT issue of overfitting to the training sample.

3.2.1.4. Multilayer perceptron

Artificial neural networks (ANNs) have widely been used in several prediction tasks in the area of finance (Kumar and Ravi, 2016). Among them, one of the simplest kinds of neural networks, and at the same time very popular is the multilayer perceptron (MLP) model. Not only for these reasons but also because MLP is able to handle all the text representations we use (TF-IDF-based or embedding-based) makes it an ideal choice for our analysis along with the rest of the machine learning models we use. In a typical MLP model, there is an input layer of neurons, where our variables, textual or financial, are used as inputs (Goldberg, 2017). Next, there are one or more hidden layers. Each neuron computes a weighted sum of its inputs, applies a non-linear activation function to the resulting sum, and passes its output to the neurons of the next layer. The weights are learned by minimizing a loss function via back-propagation, a version of stochastic gradient descent for networks with hidden layers. In a classification task, the non-linear activation functions allow the model to cope with non-linearly separable data. In binary classification, as in our case, the output layer contains a single neuron with a sigmoid activation function, which provides the probability the model assigns to the positive class. The loss function is typically binary cross-entropy, in effect minimizing the divergence of the predicted probability distribution over the two classes from the correct (one-hot) distribution, for each training example.[11]

[11] We use 5-fold cross-validation for hyper-parameter tuning. As a result, our MLP model has 3 hidden layers, each of which has 200 neurons. Given that MLP is a feed-forward model that maps inputs (financial variables and textual features) to a binary outcome (underpricing or not). Furthermore, we use Adam (a version of stochastic gradient descent) as the optimizer algorithm, and rectified linear unit (ReLU) as the activation function of each hidden layer. ReLU is defined as $f(x) = \max(0, x)$. Finally, we use early stopping to mitigate overfitting (Mai et al., 2019). To do so, we set aside 10% of training data as validation or development set.

3.2.2. MLP model with word embedding approach

In all the previous models, we use the BOW text representations as textual inputs. To utilize textual information based on word embeddings, we represent each text as the centroid of the word embeddings of the words that make it up. Next, we use these textual features as inputs to a MLP model. In this paper, we employ the MLP model with word embedding approach, either with TF or TF-IDF weighting scheme. The former model computes the average of every dimension of the word embedding vector for each word in the text, while the latter computes the weighted average based on the TF-IDF score of each word. As follows, we firstly provide the mathematical formula of the TF centroid textual feature:

$$\overrightarrow{TFcentroid_i} = \frac{\sum_j^V (TF_{ij} \times \overrightarrow{w_j})}{\sum_j^V TF_{ij}} \tag{6}$$

where i represents each text in the sample, j represents each word in the vocabulary (V), $\overrightarrow{w_j}$ represents the 200-dimensional word embedding of each word j, and TF_{ij} represents the term frequency of the word j in the text i. Moreover, we present the mathematical formula for TF-IDF centroid textual feature:

$$\overrightarrow{TF-IDFcentroid_i} = \frac{\sum_j^V (TF_{ij} \times IDF_j \times \overrightarrow{w_j})}{\sum_j^V (TF_{ij} \times IDF_j)} \tag{7}$$

where IDF_j represents the inverse document frequency of each word j.

In the MLP models with the word embedding approach, we represent each document with a (d x n) matrix, where d is the vectorized representation of each word and n refers to the document length. In practice, d is the pre-trained word embeddings, which can either be the generic word embeddings based on GloVe, or our finance word embeddings trained on the EDGAR documents.

Figure 3 illustrates the architecture of the MLP models with the word embedding approach. First, we use a 200-dimensional vector to represent each document, as the size of the pre-trained word embeddings is 200. Second, these vectors are inserted as inputs in the model, and then they are processed by two hidden layers with rectified linear unit (ReLU) activation function. Finally, there is the output layer where a sigmoid function provides the probability of the positive class.

Insert **Figure 3** here

We create our models using the Keras library with a TensorFlow backend (Chollet, 2017). We employ a batch size of 16, and the models take less than 12 epochs to converge. Finally, to control for the issue of overfitting, we use the dropout technique and the early stopping strategy, as in Mai et al. (2019).[12]

3.3. Evaluation measures

We evaluate the out-of-sample performance of our classification models using two measures. First, we use the accuracy measure, which has been extensively used in finance classification tasks (Palepu, 1986; Pasiouras et al., 2010; Mai et al., 2019). The values of the accuracy measure are in the range of [0, 1]. In our classification task, we aim to achieve accuracy scores higher than 50%, because our two datasets are fully balanced. Any score above this threshold would imply that our models yield better than chance, and vice versa. We compute *Accuracy* according to the following formula:

$$Accuracy = \frac{|TP| + |TN|}{|TP| + |FP| + |TN| + |FN|} \tag{8}$$

where *TP* is the number of observations correctly identified as bidders (or targets) by the classifier, *TN* is the number of observations correctly identified as non-involved by the classifier,

[12] The dropout method randomly omits a subset of hidden neurons at every step of the training process. On the other hand, early stopping requires monitoring the performance of the validation set, a subset of the training set, so that we stop the training process when there is no more improvement.

FP the number of observations incorrectly identified as bidders (or targets) by the classifier and *FN* is the number of observations incorrectly identified as non-involved by the classifier.

As the second step of our evaluation procedure, we employ the receiver operating characteristic (ROC) curves for the models with the highest accuracy scores in each task. ROC curves are frequently used in finance prediction tasks, such as bankruptcy prediction (Chava and Jarrow, 2004; Mai et al., 2019). The ROC curve plots the true-positive rate of the classifier on the vertical axis, and the false positive rate on the horizontal axis, as the classification threshold varies. In fact, models whose ROC curves are closer to the upper and left corner of the diagram (larger area under the curve) imply better out-of-sample classification ability. Next, we plot a 45-degree line suggesting a random assignment of class labels. Based on the ROC curve, we compute the second evaluation measure, the area under the curve (AUC). AUC values are also in the range of [0, 1]. An uninformative classifier yields a 0.5 AUC score, while 1 represents a perfect classification.

4. Empirical results and discussion

4.1. Prediction with financial variables

As the first step in our empirical analysis, we examine the predictive power of our models when we use only financial variables as inputs. In fact, we investigate whether financial variables alone can distinguish between bidders and non-involved banks and targets and non-involved banks. The results are reported in Table 3. First, we present the accuracy scores of our classification models for the bidding banks. The results indicate that financial variables have predictive power in this task, as the accuracy scores exceed the 50% threshold in all cases. In more detail, the best performing models are the RF and MLP, with accuracy scores of 62.7% and 63.6%, respectively.

Second, we present the results for the target firms. In general, accuracy scores are higher compared to what reported for the bidding banks. With the exception of SVMs, the other models achieve accuracy scores that exceed 70%. More precisely, LOGIT produces an accuracy score of 78.8%, while RF and MLP perform equally well with a score of 76.9%. These results could imply that target prediction is a more feasible task compared to bidder prediction.

Insert **Table 3** here

4.2. Prediction with textual features

In this section, we investigate whether the language used by managers in the bank annual reports has any predictive power in our merger classification task. To be consistent with our empirical setting, we will first analyze results based on the BOW approach, and then, we will report the results of the word embedding approach.

Table 4 presents out-of-sample accuracy scores of our prediction models, using only textual data as inputs based on the BOW approach. We use four different types of textual features: (1) term frequency (TF), (2) term frequency-inverse document frequency (TF-IDF), (3) term frequency with bigrams (TF + bigrams), and (4) term frequency-inverse document frequency with bigrams (TF-IDF + bigrams).[13]

Panel A of Table 4 shows the results for our first dataset (bidders/non-involved). Overall, our predictive models perform better than chance.[14] MLP yields the highest accuracy score (68.0%), followed by LOGIT (66.4%) and RF (65.1%). Notably, these scores are higher compared to the ones reported in Table 3, where we used only financial variables as inputs. This fact indicates that textual information of the 10-K filings contains vital information for predicting future acquirers

[13] Types 1 and 2 use only unigrams, and types 3 and 4 use a combination of unigrams and bigrams.
[14] The only case where we obtain an accuracy score below the 50% mark is when we use TF + bigrams as textual features in the SVM-linear model (47.8%).

in the U.S. banking industry.

Panel B of Table 4 shows the results for our second dataset (targets/non-involved). In this task, all our models achieve accuracy scores that exceed the 50% mark. In terms of model performance, MLP achieves again the highest score (84.0%), followed by LOGIT (79.5%), and RF (77.6%). The results for the target classification task are in many cases marginally better compared to the ones obtained with the use of financial variables. Therefore, we argue that textual features improve the classification ability of our models in both tasks.

Insert **Table 4** here

Table 5 reports the results when we employ textual features based on the word embedding approach. We examine the performance of two different models, the TF Centroid embedding model and the TF-IDF Centroid embedding model. In each model, we use as inputs either the generic word embeddings based on GloVe, or our finance word embeddings. More precisely, we use the MLP model with four different word embedding features: (1) TF Centroid with generic word embeddings as inputs (TF Generic centroid), (2) TF-IDF Centroid with generic word embeddings as inputs (TF-IDF Generic centroid), (3) TF Centroid with finance word embeddings as inputs (TF Finance centroid), and (4) TF-IDF Centroid with finance word embeddings as inputs (TF-IDF Finance centroid).

Panel A presents the results for the bidders and Panel B presents the results for the targets. In predicting future bidders, the TF-IDF Centroid embedding model has the best performance with both types of inputs (64.0%). In predicting future targets, the TF Centroid embedding model has the best performance, with accuracy scores of 78.0% with the use of generic word embeddings and 77.0% with the use of finance word embeddings. Taken altogether, these results suggest that textual features based on word embeddings are also meaningful inputs in our merger classification task. In fact, models using such textual data are able to predict more accurately

future bidders compared to models using only financial variables, while they have comparable performance in identifying future targets.

Insert **Table 5** here

4.3. Prediction with both financial variables and textual features

In this section, we jointly use both financial variables and textual features as inputs in our classification models. We do so, in order to investigate whether and to what extent textual information can effectively be combined with financial variables in our merger classification task.

4.3.1. Combination of financial variables with bag of words textual features

We now investigate the prediction performance when both financial variables and textual features based on BOW are utilized. One issue that emerges here is that textual features dramatically outnumber financial variables, and as a result, the plethora of textual data may overrule the role of financial variables. Such a model may suffer from the "curse of dimensionality" (Mai et al., 2019). To alleviate this concern, we have to decrease the dimensionality (number of words in the vocabulary) of our textual features.

We project our high-dimensional document vectors into a low dimensional space using the singular value decomposition (SVD) dimensionality reduction technique as in Kim et al., (2005), and Degiannakis et al. (2018), among others. In our empirical analysis, we use SVD to project the original feature vectors to 100 dimensions (SVD100).[15] In other words, this method reduces the dimensions of our textual features from 20,000 to 100. By using such a low level of textual representation, we able to deal with the curse of dimensionality, while preserving the meaningful

[15] It is worth mentioning that we take into account only the 100 first SVD components, as they were found to explain almost 80% of the joint variance of the 20,000 most frequent textual features in the 10-K filings.

information of the 10-K filings.[16]

Table 6 presents the results of this analysis. Panel A reports the accuracy scores for our first dataset (bidders/non-involved). First, RF is the best performing model with an accuracy score of 67.2%. This score is achieved with the combination of financial variables and TF-IDF$_{SVD100}$ textual features. Next, MLP produces the second best accuracy score (65.1%), followed by LOGIT (64.3%). Finally, SVMs (both linear and RBF) have the poorest performance, since their accuracy scores range between 45.0% and 57.9%.

Two inferences are obtained when we compare the results of the models using both types of inputs with the models using a single type of input. On the one hand, the performance of our models is substantially improved when we use both textual features and financial variables instead of only financial variables. On the other hand, the performance of the former set of models is comparable to the performance of the models using only textual features as inputs. Collectively, these findings may indicate that textual information of the bank annual reports is more informative relative to financial variables in the bidder prediction task.

Panel B of Table 6 reports the accuracy scores for our second dataset (targets/non-involved). Interestingly, the results indicate that in the case of targets, the combination of textual features with financial variables yields the best performance so far. Strikingly, accuracy scores exceed 80% in many cases. Again, RF outperforms the other models, achieving an accuracy score of 89.7% with the use of (TF-IDF + bigrams)$_{SVD100}$. MLP comes second best with an accuracy score of 80.7%, when we use (TF-IDF + bigrams)$_{SVD100}$ as textual features. LOGIT also performs reasonably well, as its best score equals 74.4%. In line with previous findings, SVMs have the lowest performance, as their accuracy scores range between 61.5% and 69.9%. Overall, the

[16] We present results without the SVD technique in Table A2 of the Appendices. In the bidder prediction task, the performance of our models is comparable with the ones using the SVD100 textual features. However, in the case of targets, our models performance is in general lower.

performance of our models in predicting future targets is substantially improved when we use both types of inputs than when we use a single type of input.[17] These results indicate that in this task, textual information can effectively be combined with financial variables to produce more accurate estimates.

Insert **Table 6** here

4.3.2. Combination of financial variables with word embedding textual features

In this section, we examine the out-of-sample performance of the MLP model using a combination of word embedding textual features and financial variables as inputs. Table 7 presents our findings for bidder classification (Panel A) and target classification (Panel B).

The results of Panel A suggest that the combination of word embeddings with financial variables provides the most accurate estimates for our bidder classification task. More precisely, the TF-IDF Finance centroid achieves an accuracy score of 72.0%. This score is substantially higher compared to what reported in previous tables. In addition, even our lowest score (67.0%), which is produced by the TF Generic centroid, is higher than the vast majority of scores produced by other models (either with a single type of input or both types of inputs).

Further, our results provide two additional important findings. First, the TF-IDF centroid embedding model outperforms the TF centroid embedding model. This means that the TF-IDF weighting scheme produces a set of weights for our textual features that enhance the ability of our models to classify bidders from non-involved banks. This result is consistent with previous findings, as the TF-IDF approach tends to perform better in many NLP tasks compared to simple proportional weighting (Loughran and McDonald, 2011; Loughran and McDonald, 2016; Katsafados et al., 2020). Second, the use of our finance word embeddings increases the

[17] We also re-run our models using as inputs only the SVD100 textual features (see Table A3 in the Appendices). The main inferences of this analysis remain the same.

performance of both the TF and the TF-IDF centroid embedding model, compared to using generic word embeddings. Therefore, we argue that the finance word embeddings are more meaningful inputs than generic word embeddings in a bidder classification task. This is expected to some extent, because FWEs take into account the most likely meaning of a word in a business context, and as such, they are able to understand better the semantics of the text.

The results of Panel B of Table 7 indicate that the combination of financial variables with word embeddings performs well also in the target classification task. In fact, our models provide accuracy scores in the range of 78.0% to 81.0%. First, these findings suggest that textual features based on word embeddings can be effectively combined with financial variables in identifying future targets. The aforementioned accuracy scores are in general higher compared to the ones reported when we used either financial variables or textual data as separate inputs in our machine learning models. Second, our finance word embeddings produce again better estimates compared to the generic ones. Finally, it is worth mentioning that we have achieved even higher accuracy scores in this task using machine learning models (especially with the RF model). However, the accuracy scores of around 80% are high enough to suggest that the MLP models with the word embedding approach provide reasonably accurate estimates in the target prediction task.

Insert **Table 7** here

4.3.3. ROC curves

We further examine the predictive ability of our models using the ROC curves. In the bidder classification task, we construct the ROC curves for the TF and the TF-IDF centroid embedding models, as those were the best performing models in this task. In all models, we jointly use financial variables and textual features as inputs. Notably, in all cases, the AUC values are higher than 0.7, as we observe in Figure 4. To begin with, when we use the generic word embeddings, the AUC score is 0.70 for the TF Generic centroid model and 0.74 for the TF-IDF Generic

centroid model. Further, in line with our previous findings, AUC scores increase when we use our finance word embeddings. More precisely, the AUC score is 0.71 for the TF Finance centroid model and 0.76 for the TF-IDF Finance centroid model. In sum, our results suggest that the TF-IDF weighting scheme offers better predictive power, while the use of our finance word embeddings enhances the classification ability of the MLP models with the word embedding approach.

Insert **Figure 4** here

Figure 5 depicts the ROC curves for the three best performing machine learning models based on BOW features in the target prediction task. In practice, these models are the same as the ones in Table 6, where we used both types of inputs (textual and financial data), and four different types of textual features. When it comes to model performance, RF yields the most accurate estimates, as AUC values range from 0.90 to 0.93. MLP comes next with AUC values in the range of 0.80 to 0.88, and finally, LOGIT with values from 0.76 to 0.82. Remarkably, all three models achieve their highest AUC score with the use of $(\text{TF-IDF + bigrams})_{SVD100}$ textual features. Overall, these results support our previous findings.

Insert **Figure 5** here

4.3.4. Bootstrap statistical significance test

So far, our two performance measures have provided robust evidence regarding the performance of our models. In the bidder classification task, the best performing model is the TF-IDF Finance centroid, where in the target classification task, the RF model with the use of the $(\text{TF-IDF + bigrams})_{SVD100}$ yields the more accurate estimates. However, it is important to test the consistency of these results, by including statistical significance tests to validate metric gains. To do so, we employ the bootstrap resampling method of Berg-Kirkpatrick et al. (2012). We provide a more detailed description of this technique in Appendix B.

In Panel A of Table 8, we compare the TF-IDF Finance centroid with the other MLP models based on the word embedding approach. The comparisons suggest that the TF-IDF Finance centroid significantly outperforms the TF-IDF Generic centroid (p=0.006) and the TF Generic centroid (p=0.000). Further, the TF Finance centroid significantly outperforms the TF Generic centroid (p=0.000). Then, we compare the performance of the TF-IDF Generic centroid against the TF Finance centroid and the TF Generic centroid. Interestingly, the TF-IDF Generic centroid significantly outperforms only the TF Generic centroid (p=0.049). Collectively, these findings provide strong evidence that the TF-IDF weighting scheme provides better results compared to TF, and our finance word embeddings have higher information content relative to generic word embeddings.

In Panel B of Table 8, we compare the following four models: (1) RF with the use of TF_{SVD100} (RF-TF_{SVD100}), (2) RF with the use of TF-IDF_{SVD100} (RF-TF-IDF_{SVD100}), (3) RF with the use of $(TF + bigrams)_{SVD100}$ (RF-$(TF + bigrams)_{SVD100}$), and (4) RF with the use of $(TF\text{-}IDF + bigrams)_{SVD100}$ (RF-$(TF\text{-}IDF + bigrams)_{SVD100}$). In particular, we compare the RF-$(TF\text{-}IDF + bigrams)_{SVD100}$ with the remaining three models. In all three comparisons, the p-value equals 0.000, which suggests that the RF-$(TF\text{-}IDF + bigrams)_{SVD100}$ significantly outperforms all the other RF models. Also, RF-TF-IDF_{SVD100} appears to be the second-best performing model, since it outperforms RF-$(TF + bigrams)_{SVD100}$ (p=0.036) and RF-TF_{SVD100} (p=0.014). Yet the performance of the RF-$(TF + bigrams)_{SVD100}$ is comparable with the RF-TF_{SVD100} (p=0.684). In summary, TF-IDF weight enhances the overall performance of the models, while the combination of unigrams and bigrams is meaningful when we use the TF-IDF weighting scheme.

Insert **Table 8** here

4.3.5. Additional robustness tests

Our examination period includes some special years that may have an impact on our results, such as the years of the dot-com bubble, and the years of the financial crisis (Cohen, 2020). To ensure that our results are not impacted by these time periods, we remove all bank-year observations from years 2000-2001 as the years of the dot-com bubble, and from years 2008-2009 as the years of the financial crisis. Then, we repeat the analysis of Tables 6 and 7. Our results remain qualitatively similar (see Tables A4 and A5 in the Appendices). Furthermore, we examine whether the results of Table 6 are sensitive to the dimensions of our textual features. To do so, we lower the threshold of most frequent words from 20,000 to 10,000. Our results remain again similar to the ones reported in our baseline analysis (see Table A6 in the Appendices).

To further illustrate the high importance of textual features, we adopt the Gini impurity technique (Kurt et al., 2008). Practically, this technique computes the importance score for each variable in the model, and it is applied to the RF models. Hence, we compute the Gini importance scores for the 20 most important features of our RF models. We limit the analysis to the 20 most important features, due to the fact that our textual features substantially outnumber our financial variables. Then, we compute the sum of these scores separately for textual features and for financial variables. By comparing those sums, we observe that textual features are more important inputs than financial variables in all cases and by a large margin (see Table A7 in the Appendices).

5. Conclusions

In this study, we utilize several machine learning models to predict bank mergers in the U.S. Our key innovation is that we investigate the role of textual disclosure of bank annual reports in our merger prediction task. More precisely, we examine whether the language used by bank

managers in the annual reports has any additional predictive power in our classification models beyond the traditional financial variables. The intuition behind this text-based approach is that textual information could reduce the opaqueness of bank assets and provide some important insights regarding the strategic options of the banking firms. Hence, our study contributes to the recent body of research that utilizes textual analysis in various finance tasks.

We create a comprehensive dataset of 9,207 U.S. bank-year observations during the period 1994-2016. To create our textual features, we use the bag of words and the word embedding approaches. One important aspect of our empirical approach is that we go beyond the frequently-used generic word embeddings, and we create our own word embeddings specialized in the finance sector. Then, we use our textual features (with or without financial variables) as inputs in our classification models, and we evaluate the models' out-of-sample performance according to the accuracy measure.

Our findings provide strong evidence for the importance of textual information in a bank merger classification task. First, when we use a single type of input (textual data or financial variables), we observe that models using textual data provide better, or at least similar, accuracy scores compared to models using only financial variables. Second, when we jointly use both types of inputs, the out-of-sample performance is substantially improved. However, the best performing model is different in each task. In the bidder classification task, the MLP model with the finance word embeddings achieves the highest accuracy score of 72%. It is noteworthy that the MLP models with the word embedding approach generally produce the highest scores with our finance word embeddings compared with the generic ones. In the target prediction task, the TF-IDF weighted RF model using a combination of unigrams and bigrams achieves the highest score of 89.7%. In addition to accuracy scores, we also generate the ROC curves and evaluate the

performance of our models using the AUC scores. Notably, our inferences remain the same. Finally, our results are robust to a series of robustness tests.

To conclude, this paper is the first that highlights the utility of textual information in a bank merger prediction task. We argue that the use of text is detrimental in the bank merger prediction due to the inherently opaque nature of the baking industry. Further, we introduce new methodological insights on how textual features can effectively be combined with financial variables in machine learning models to produce better out-of-sample performance. For these reasons, we hope that our study will provide fertile ground for future research in the fast-growing literature of textual analysis in finance.

References

Ahmed, A. S., Takeda, C., & Thomas, S. (1999). Bank loan loss provisions: A reexamination of capital management, earnings management and signaling effects. *Journal of Accounting and Economics*, 28, 1-25.

Ambrose, B. W., & Megginson, W. L. (1992). The role of asset structure, ownership structure, and takeover defenses in determining acquisition likelihood. *Journal of Financial and Quantitative Analysis*, 27, 575-589.

Baele, L., De Bruyckere, V., De Jonghe, O., & Vander Vennet, R. (2015). Model uncertainty and systematic risk in US banking. *Journal of Banking and Finance*, 53, 49-66.

Balakrishnan, R., Qiu X. Y., & Srinivasan, P. (2010). On the predictive ability of narrative disclosures in annual reports. *European Journal of Operational Research*, 202, 789-801.

Barnes, P. (1998). Can takeover targets be identified by statistical techniques? Some UK evidence. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 47, 573-591.

Barnes, P. (1999). Predicting UK takeover targets: Some methodological issues and an empirical study. *Review of Quantitative Finance and Accounting*, 12, 283-302.

Beatty, A. L., Ke, B., & Petroni, K. R. (2002). Earnings management to avoid earnings declines across publicly and privately held banks. *Accounting Review*, 77, 547-570.

Becher, D. A. (2009). Bidder returns and merger anticipation: evidence from banking deregulation. *Journal of Corporate Finance*, 15, 85-98.

Berg-Kirkpatrick, T., Burkett, D., & Klein, D. (2012). An empirical investigation of statistical significance in nlp. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (pp. 995-1005).

Blau, B. M., Brough, T. J., & Griffith, T. G. (2017). Bank opacity and the efficiency of stock prices. *Journal of Banking and Finance*, 76, 32-47.

Bodnaruk, A., Loughran, T., & McDonald, B. (2015). Using 10-K text to gauge financial constraints. *Journal of Financial and Quantitative Analysis*, 50, 623-646.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24, 123-140.

Breiman, L. (2001). Random forests. *Machine Learning*, 45, 5-32.

Brook, Y., Hendershott, R., & Lee, D. (1998). The gains from takeover deregulation: Evidence from the end of interstate banking restrictions. *Journal of Finance*, 53, 2185-2204.

Brown, S. V, & Tucker, J. W. (2011). Large-sample evidence on firms' year-over-year MD&A modifications. *Journal of Accounting Research*, 49, 309-346.

Bushman, R. M., & Williams, C. D. (2012). Accounting discretion, loan loss provisioning, and discipline of banks' risk-taking. *Journal of Accounting and Economics*, 54, 1-18.

Cao, L. (2003). Support vector machines experts for time series forecasting. *Neurocomputing*, 51, 321-339.

Chava, S., & Jarrow, R. A. (2004). Bankruptcy prediction with industry effects. *Review of Finance*, 8, 537-569.

Chollet, F. (2017). Deep learning with Python. Manning Publications.

Cohen, L., Malloy, C., & Nguyen, Q. (2020). Lazy prices. *Journal of Finance*, 75, 1371-1415.

Comment, R., & Schwert, G. W. (1995). Poison or placebo? Evidence on the deterrence and wealth effects of modern antitakeover measures. *Journal of Financial Economics*, 39, 3-43.

Cornett, M. M., McNutt, J. J., & Tehranian, H. (2006). Performance changes around bank mergers: Revenue enhancements versus cost reductions. *Journal of Money, Credit and Banking*, 38, 1013-1050.

Cornett, M. M., & Tehranian, H. (1992). Changes in corporate performance associated with bank acquisitions. *Journal of Financial Economics*, 31, 211-234.

Cremers, K. J. M., Nair, V. B., & John, K. (2009). Takeovers and the cross-section of returns. *Review of Financial Studies*, 22, 1409-1445.

Degiannakis, S., Filis, G., & Hassani, H. (2018). Forecasting global stock market implied volatility indices. *Journal of Empirical Finance*, 46, 111-129.

DeLong, G.L., & DeYoung, R. (2007). Learning by observing: information spillovers in the execution and valuation of commercial bank M&As. *Journal of Finance*, 62,181-216.

Demirgüç-Kunt, A., & Huizinga, H. (2013). Are banks too big to fail or too big to save? International evidence from equity prices and CDS spreads. *Journal of Banking and Finance*, 37, 875-894.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Doumpos, M., Andriosopoulos, K., Galariotis, E., Makridou, G., & Zopounidis, C. (2017). Corporate failure prediction in the European energy sector: A multicriteria approach and the effect of country characteristics. *European Journal of Operational Research*, 262, 347-360.

Edmans, A., Goldstein, I., & Jiang, W. (2012). The real effects of financial markets: The impact of prices on takeovers. *Journal of Finance*, 67, 933-971.

Ellul, A., & Yerramilli, V. (2013). Stronger risk controls, lower risk: Evidence from US bank holding companies. *Journal of Finance*, 68, 1757-1803.

Espahbodi, H., & Espahbodi, P. (2003). Binary choice models and corporate takeover. *Journal of Banking and Finance,* 27, 549-574.

Filson, D., & Olfati, S. (2014). The impacts of Gramm–Leach–Bliley bank diversification on value and risk. *Journal of Banking and Finance*, 41, 209-221.

Flannery, M. J., Kwan, S. H., & Nimalendran, M. (2004). Market evidence on the opaqueness of banking firms' assets. *Journal of Financial Economics*, 71, 419-460.

Flannery, M. J., Kwan, S. H., & Nimalendran, M. (2013). The 2007–2009 financial crisis and bank opaqueness. *Journal of Financial Intermediation*, 22, 55-84.

Gandhi, P., Loughran, T., & McDonald, B. (2019). Using annual report sentiment as a proxy for financial distress in US banks. *Journal of Behavioral Finance*, 20, 424-436.

Geng, R., Bose, I., & Chen, X. (2015). Prediction of financial distress: An empirical study of listed Chinese companies using data mining. *European Journal of Operational Research*, 241, 236-247.

Goldberg, Y. (2017). Neural Network Methods for Natural Language Processing. Morgan & Claypool Publishers.

Hankir, Y., Rauch, C., & Umber, M. P. (2011). Bank M&A: A market power story? *Journal of Banking and Finance*, 35, 2341-2354.

Hanley, K. W., & Hoberg, G. (2010). The information content of IPO prospectuses. *Review of Financial Studies*, 23, 2821-2864.

Hannan, T. H., & Rhoades, S. A. (1987). Acquisition targets and motives: The case of the banking industry. *Review of Economics and Statistics*, 69, 67-74.

Hasbrouck, J. (1985). The characteristics of takeover targets q and other measures. *Journal of Banking and Finance*, 9, 351-362.

Houston, J. F., James, C. M., & Ryngaert, M. D. (2001). Where do merger gains come from? Bank mergers from the perspective of insiders and outsiders. *Journal of Financial Economics*, 60, 285-331.

Huizinga, H., & Laeven, L. (2012). Bank valuation and accounting discretion during a financial crisis. *Journal of Financial Economics*, 106, 614-634.

Imbens, G. W. (1992). An efficient method of moments estimator for discrete choice models with choice-based sampling. *Econometrica*, 60, 1187-1214.

Jegadeesh, N., & Wu, D. (2013). Word power: A new approach for content analysis. *Journal of Financial Economics*, 110, 712-729.

Jiang, L., Levine, R., & Lin, C. (2016). Competition and bank opacity. *Review of Financial Studies*, 29, 1911-1942.

Jones, J. S., Lee, W. Y., & Yeager, T. J. (2013). Valuation and systemic risk consequences of bank opacity. *Journal of Banking and Finance*, 37, 693-706.

Katsafados, A. G., Androutsopoulos, I., Chalkidis, I., Fergadiotis, E., Leledakis, G. N., & Pyrgiotakis, E. G. (2021). Using textual analysis to identify merger participants: Evidence from U.S. banking industry. *Finance Research Letters,* Forthcoming.

Katsafados, A. G., Androutsopoulos, I., Chalkidis, I., Fergadiotis, E., Leledakis, G. N., & Pyrgiotakis, E. G. (2020). Textual information and IPO underpricing: A machine learning approach. Working Paper, Available at SSRN: https://ssrn.com/abstract=3720213

Kearney, C., & Liu, S. (2014). Textual sentiment in finance: A survey of methods and models. *International Review of Financial Analysis*, 33, 171-185.

Kim, H., Howland, P., & Park, H. (2005). Dimension reduction in text classification with support vector machines. *Journal of Machine Learning Research*, 6, 37-53.

Kumar, B. S., & Ravi, V. (2016). A survey of the applications of text mining in financial domain. *Knowledge-Based Systems*, 114, 128-147.

Kurt, I., Ture, M., & Kurum, A. T. (2008). Comparing performances of logistic regression, classification and regression tree, and neural networks for predicting coronary artery disease. *Expert Systems with Applications*, 34, 366-374.

Laitinen, T., & Kankaanpaa, M. (1999). Comparative analysis of failure prediction methods: The Finnish case. *European Accounting Review*, 8, 67-92.

Leledakis, G. N., & Pyrgiotakis, E. G. (2021). U.S. bank M&As in the post-Dodd–Frank Act era: Do they create value? *Journal of Banking and Finance*, Forthcoming.

Leledakis, G. N., Mamatzakis, E. C., Pyrgiotakis, E. G., & Travlos, N. G. (2021). Does it pay to acquire private firms? Evidence from the US banking industry. *European Journal of Finance*, Forthcoming.

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

Loughran, T., & McDonald, B. (2013). IPO First-day returns, offer price revisions, volatility, and form S-1 language. *Journal of Financial Economics*, 109, 307-326.

Loughran, T., & McDonald, B. (2014). Measuring readability in financial disclosures. *Journal of Finance*, 69, 1643-1671.

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

Mai, F., Tian, S., Lee, C., & Ma, L. (2019). Deep learning models for bankruptcy prediction using textual disclosures. *European Journal of Operational Research*, 274, 743-758.

Manne, H. G. (1965). Mergers and the market for corporate control. *Journal of Political Economy*, 73, 110-120.

Manning, C., & Schutze, H. (1999). Foundations of statistical natural language processing. The MIT Press, Cambridge, US.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26, 3111-3119.

Min, J. H., & Lee, Y. C. (2005). Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. *Expert Systems with Applications*, 28, 603-614.

Morgan, D. P. (2002). Rating banks: Risk and uncertainty in an opaque industry. *American Economic Review*, 92, 874-888.

Nassirtoussi, A. K., Aghabozorgi, S., Wah, T. Y., & Ling Ngo, D. C. (2014). Text mining for market prediction: A systematic review. *Expert Systems with Applications*, 41, 7653-7670.

Neophytou, E., & Molinero, C. M. (2004). Predicting corporate failure in the UK: A multidimensional scaling approach. *Journal of Business Finance and Accounting*, 31, 677-710.

O'hara, M., & Shaw, W. (1990). Deposit insurance and wealth effects: the value of being "too big to fail". *Journal of Finance*, 45, 1587-1600.

Pai, P. F., & Lin, C. S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. *Omega*, 33, 497-505.

Palepu, K. G. (1986). Predicting takeover targets: A methodological and empirical analysis. *Journal of Accounting and Economics*, 8, 3-35.

Pasiouras, F., Gaganis, S., & Zopounidis, C. (2010). Multicriteria classification models for the identification of targets and acquirers in the Asian banking sector. *European Journal of Operational Research*, 204, 328-335.

Pasiouras, F., Tanna, S., & Zopounidis, C. (2007). The identification of acquisition targets in the EU banking industry: An application of multicriteria approaches. *International Review of Financial Analysis*, 16, 262-281.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In Empirical Methods in Natural Language Processing (pp. 1532-1543).

Powell, R. G. (2001). Takeover prediction and portfolio performance: A note. *Journal of Business Finance and Accounting*, 28, 993-1011.

Prasad, R. M., & Melnyk, Z. L. (1991). Positioning banks for acquisitions: A research note. *Economics Letters*, 35, 51-56.

Ramaswamy, K. (1997). The performance impact of strategic similarity in horizontal mergers: Evidence from the US banking industry. *Academy of Management Journal*, 40, 697-715.

Rhoades, S. A. (1993). Efficiency effects of horizontal (in-market) bank mergers. *Journal of Banking and Finance*, 17, 411-422.

Rhoades, S. A. (1998). The efficiency effects of bank mergers: An overview of case studies of nine mergers. *Journal of Banking and Finance*, 22, 273-291.

Rogers, J. L., Van Buskirk, A., & Zechman, S. L. C. (2011). Disclosure tone and shareholder litigation. *Accounting Review*, 86, 2155-2183.

Routledge, B. R., Sacchetto, S., & Smith, N. A. (2017). Predicting merger targets and acquirers from text. Working Paper, Carnegie Mellon University.

Shin, K. S., Lee, T. S., & Kim, H. (2005). An application of support vector machines in bankruptcy prediction model. *Expert Systems with Applications*, 28, 127-135.

Slowinski, R., Zopounidis, C., & Dimitras, A. I. (1997). Prediction of company acquisition in Greece by means of the rough set approach. *European Journal of Operational Research*, 100, 1-15.

Thompson, S. (1997). Takeover activity among financial mutuals: An analysis of target characteristics. *Journal of Banking and Finance*, 21, 37-53.

Vapnik, V. (1998). Statistical learning theory. (1st ed.). New York: Wiley.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems, (pp 6000-6010).

Veganzones, D., & Severin, E. (2018). An investigation of bankruptcy prediction in imbalanced datasets. *Decision Support Systems*, 112, 111-124.

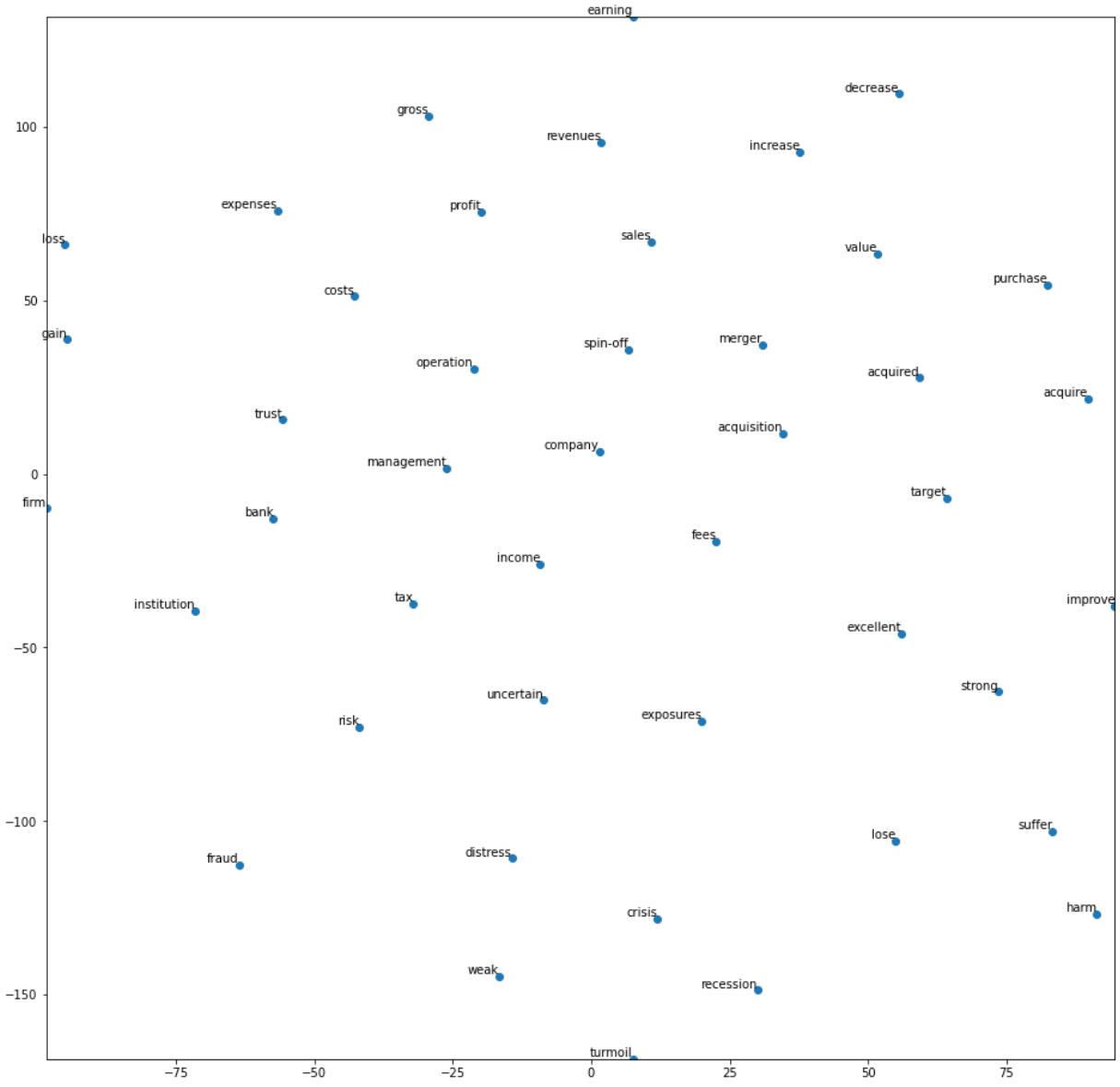
Wheelock, D. C., & Wilson, P. W. (2000). Why do banks disappear? The determinants of US bank failures and acquisitions. *Review of Economics and Statistics*, 82, 127-138.

Wu, C. H., Tzeng, G. H., Goo, Y. J., & Fang, W. C. (2007). A real-valued genetic algorithm to optimize the parameters of support vector machine for predicting bankruptcy. *Expert Systems with Applications*, 32, 397-408.

Zaheer, M., Guruganesh, G., Dubey, A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., & Ahmed, A. (2020). Big bird: Transformers for longer sequences. arXiv preprint arXiv:2007.14062.

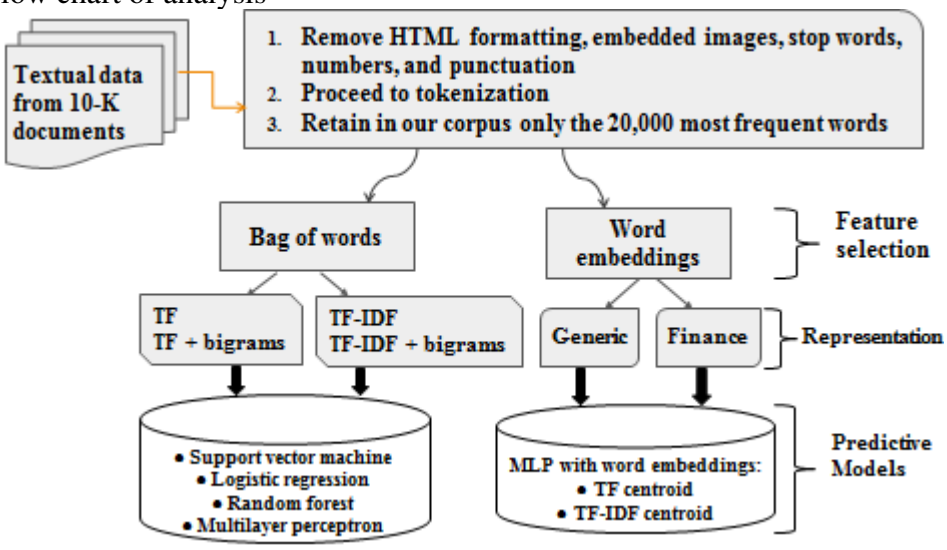
Zheng, Y. (2020). Does bank opacity affect lending? *Journal of Banking and Finance*, 119, 105900.

Figure 1
2-dimensional representation of finance word embeddings



This figure visualizes the position of various words from our finance word embeddings (FWE) into a 2-dimensional vector space. To reduce the dimensions of word embeddings from 200 to 2, we use the t-Distributed Stochastic Neighbor Embedding (t-SNE) dimensionality reduction technique. Words that share morpho-syntactic or semantic properties are mapped in close proximity to each other in the word embedding space.

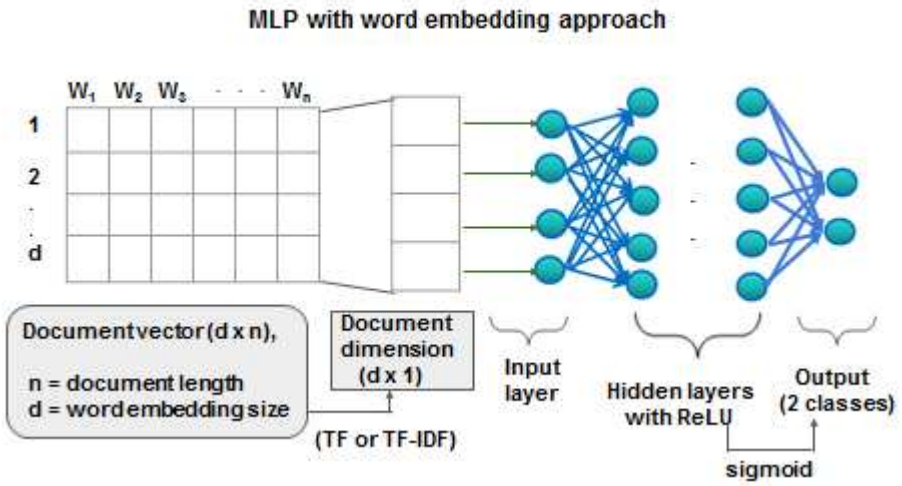
Figure 2
Flow chart of analysis



This figure describes our textual analysis process step by step, including textual data collection, preprocessing, feature selection, feature representation and model evaluation.

Figure 3

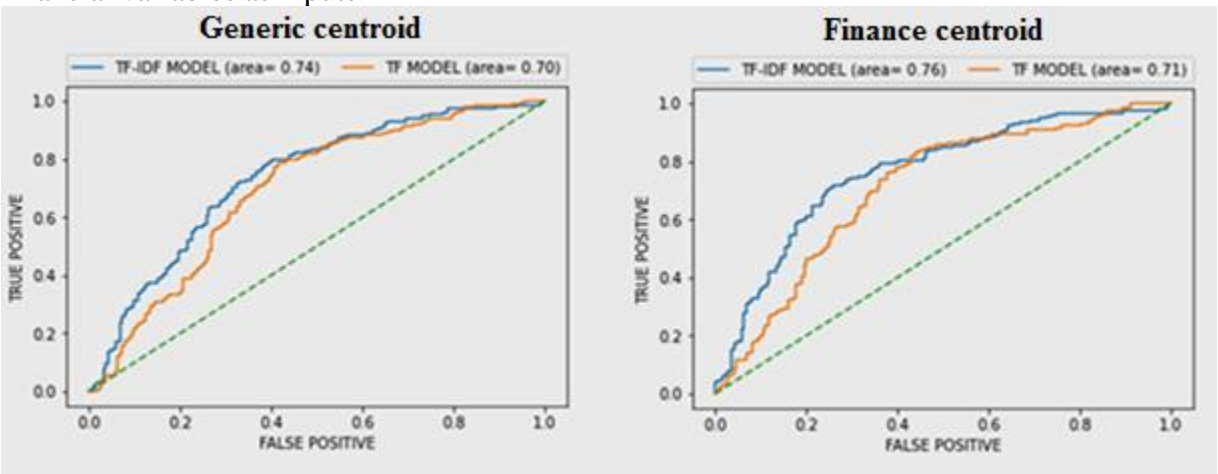
Architecture of the MLP models with the word embedding approach



This figure illustrates the architecture of the MLP model with word embedding approach. Sigmoid refers to the $f(x) = \frac{e^x}{1+e^x}$ and ReLU represents the $f(x) = \max(0, x)$. The spheres stand for each neuron of the neural network, and TF-IDF for the term frequency-inverse document frequency scheme.

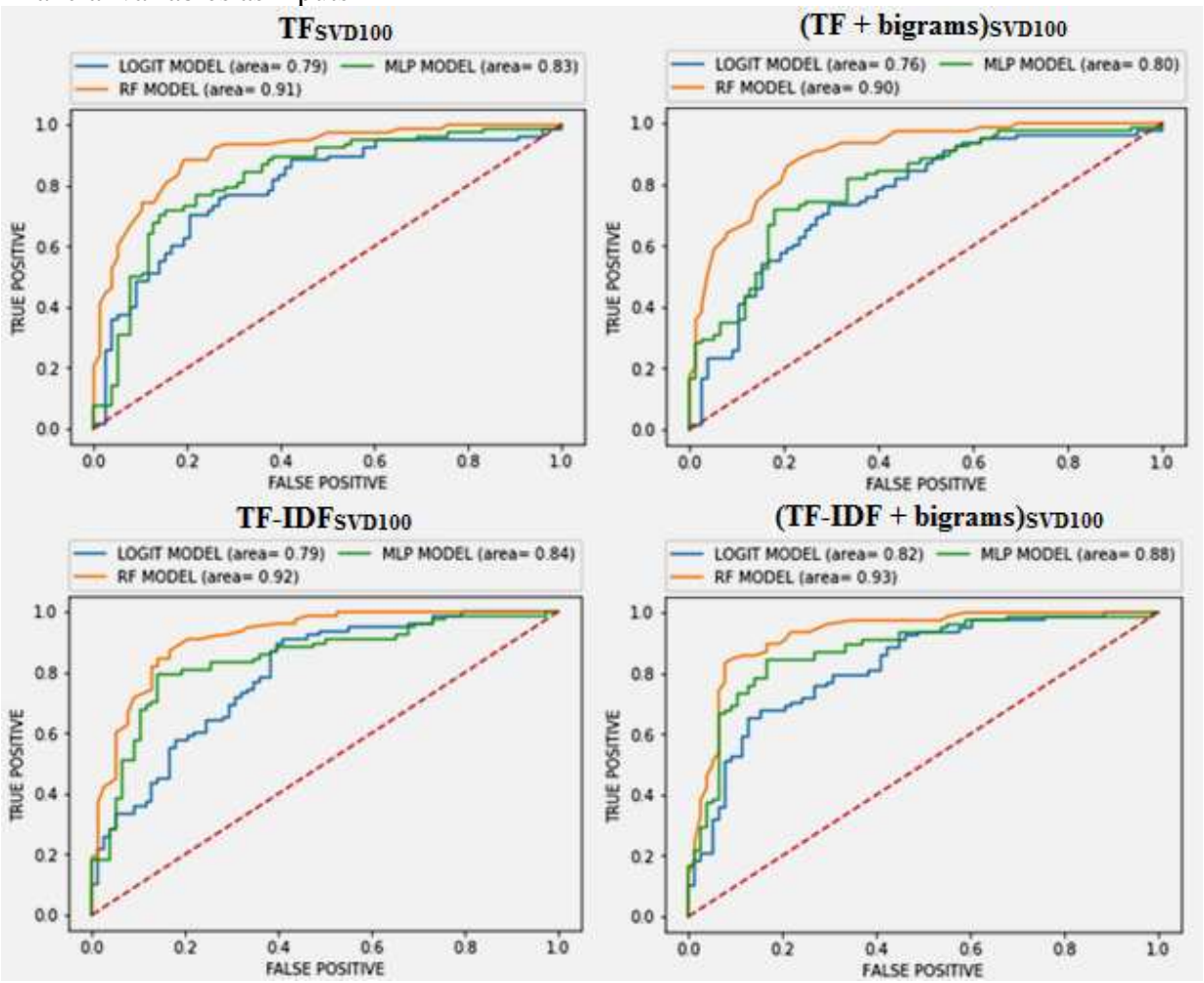
Figure 4

ROC curve of bidders using both textual features based on the word embedding approach and financial variables as inputs



This figure depicts the receiver operating characteristic (ROC) curves of bidders for the MLP model using both textual features and financial variables as inputs. We use the MLP model with four different word embedding features: (1) TF Centroid embedding model with generic word embeddings (TF Generic centroid), (2) TF-IDF Centroid embedding model with generic word embeddings (TF-IDF Generic centroid), (3) TF Centroid embedding model with finance word embeddings (TF Finance centroid), and (4) TF-IDF Centroid embedding model with finance word embeddings (TF-IDF Finance centroid). The dotted line represents a 45-degree line which indicates a random assignment of class labels. Area stands for the area under curve (AUC) measure. TF and TF-IDF represent the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Figure 5
ROC curve of targets using both SVD100 textual features based on bag of words approach and financial variables as inputs



This figure depicts the receiver operating characteristic (ROC) curves of targets for three machine learning models: (1) logistic regression (LOGIT), (2) random forest (RF), and (3) multilayer preceptor (MLP). The dotted line represents a 45-degree line which indicates a random assignment of class labels. Area stands for the area under curve (AUC) measure. TF and TF-IDF represent the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme. Bigrams are word pairs represented as a single textual feature. The figures on the left hand side report results using only unigrams, while the figures on the right hand side report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature.

Table 1

Yearly distribution of observations

Filing year	Bidders	Targets	Non-involved	All
1994	39	5	76	120
1995	39	11	120	167
1996	55	14	295	363
1997	84	33	382	494
1998	81	30	396	504
1999	49	24	441	513
2000	43	29	438	509
2001	35	21	430	486
2002	35	17	429	481
2003	66	32	402	496
2004	48	16	449	512
2005	57	18	417	492
2006	57	25	391	471
2007	37	13	353	403
2008	15	8	360	383
2009	4	3	366	373
2010	10	11	339	360
2011	11	7	323	341
2012	29	10	311	350
2013	40	16	319	375
2014	48	16	301	364
2015	52	17	283	352
2016	32	13	253	298
Total	966	389	7,874	9,207

This table summarizes the yearly distribution of our sample based on 10-K filing year. *Bidders* is the number of banks that participate in a merger with the role of bidder within a twelve-month period after the 10-K filing date. Similarly, *Targets* is the number of banks that participate in a merger with the role of target within a twelve-month period after the 10-K filing date, and *Non-involved* is the number of banks that were not involved in a merger in that period. *All* represents the total number of bank-year observations per filing year. Note that *All* is not always the sum of three previous categories, because a bank it is possible to have the role of both bidder and target within the twelve-month period after the filing date.

Table 2

Summary statistics

Variables	N	Mean	Median	Std. Dev.
Panel A: Bidders				
Cost efficiency %	966	63.15	63.09	10.37
ROA %	966	1.07	1.08	0.48
Size	966	15.19	15.04	1.60
Capital strength %	966	9.58	9.24	2.37
Loans %	966	65.97	66.93	10.17
Market power %	966	0.35	0.06	1.08
Asset quality %	966	0.34	0.28	0.47
Non-interest income %	966	24.90	23.58	11.99
Deposits %	966	76.22	77.87	8.66
Panel B: Targets				
Cost efficiency %	389	67.88	66.42	15.66
ROA %	389	0.82	0.91	0.79
Size	389	14.24	13.89	1.50
Capital strength %	389	9.28	8.69	2.94
Loans %	389	66.56	67.17	10.23
Market power %	389	0.17	0.02	0.65
Asset quality %	389	0.43	0.25	0.86
Non-interest income %	389	20.74	18.50	11.82
Deposits %	389	77.31	78.61	9.64
Panel C: Non-involved				
Cost efficiency %	7,874	68.16	65.32	21.60
ROA %	7,874	0.73	0.91	1.08
Size	7,874	14.24	13.89	1.56
Capital strength %	7,874	9.34	9.00	2.87
Loans %	7,874	66.25	67.36	12.17
Market power %	7,874	0.21	0.02	1.22
Asset quality %	7,874	0.58	0.29	1.00
Non-interest income %	7,874	22.22	20.19	13.59
Deposits %	7,874	77.11	79.21	10.30
Panel D: All				
Cost efficiency %	9,207	67.63	65.02	20.56
ROA %	9,207	0.77	0.94	1.03
Size	9,207	14.34	13.98	1.59
Capital strength %	9,207	9.37	9.02	2.83
Loans %	9,207	66.24	67.30	11.90
Market power %	9,207	0.22	0.02	1.19
Asset quality %	9,207	0.55	0.29	0.96
Non-interest income %	9,207	22.44	20.44	13.40
Deposits %	9,207	77.03	79.04	10.12

This table reports the summary statistics of our final (imbalanced) sample. In fact, we split the sample into the following categories: bidders (Panel A), targets (Panel B), non-involved (Panel C), and all (Panel D). The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. *Cost efficiency* is the cost to income ratio. *ROA* is calculated as the net income divided by the total assets. *Size* is the natural logarithm of the total assets. Note that total assets are measured in thousands of U.S. dollars. *Capital strength* is the ratio of common equity to the total assets. *Loans* is the ratio of loans to total assets. *Market power* is the ratio of each bank's deposits to the total deposits of the U.S. banking sector at a specific year. *Asset quality* is the amount of loan loss provisions divided by the total assets. *Non-interest income* is the ratio of non-interest income to the total income. *Deposits* is the ratio of deposits to the total assets. See Table A1 in the Appendices for the corresponding codes from the FR Y-9C and the Call reports. Note that *All* is not always the sum of three previous categories, because a bank it is possible to have the role of both bidder and target within the twelve-month period after the filing date.

Table 3

Out-of-sample performance using only financial variables

	SVM-linear	SVM-RBF	LOGIT	RF	MLP
Bidders	0.550	0.555	0.534	0.627	0.636
Targets	0.506	0.557	0.788	0.769	0.769

This table reports the accuracy scores for our machine learning models, using financial variables as inputs. The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). The analysis for bidders is based on a balanced sample of 966 bidders and 966 non-involved banks. The analysis for targets is based on a balanced sample of 389 targets and 389 non-involved banks. We use the following machine learning models: support vector machines (SVM-linear), support vector machines with radial basis function kernel (SVM-RBF), logistic regression (LOGIT), random forest (RF), and multilayer perceptron (MLP).

Table 4

Out-of-sample performance using only textual features based on bag of words approach

	SVM-linear	SVM-RBF	LOGIT	RF	MLP
Panel A: Bidders					
TF	0.532	0.501	0.664	0.651	0.669
TF-IDF	0.512	0.568	0.636	0.651	0.664
TF + bigrams	0.478	0.504	0.659	0.610	0.661
TF-IDF + bigrams	0.568	0.568	0.649	0.649	0.680
Panel B: Targets					
TF	0.513	0.571	0.731	0.731	0.808
TF-IDF	0.564	0.558	0.795	0.769	0.833
TF + bigrams	0.558	0.583	0.737	0.705	0.801
TF-IDF + bigrams	0.756	0.532	0.750	0.776	0.840

This table reports the accuracy scores for our machine learning models, using textual features based on the bag of words approach. The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. The first two lines of each panel report results using only unigrams, while the last two lines report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature. We use the following machine learning models: support vector machines (SVM-linear), support vector machines with radial basis function kernel (SVM-RBF), logistic regression (LOGIT), random forest (RF), and multilayer perceptron (MLP). TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table 5

Out-of-sample performance of the MLP model using only textual features based on word embedding approach

	Generic centroid	Finance centroid
Panel A: Bidders		
TF	0.510	0.540
TF-IDF	0.640	0.640
Panel B: Targets		
TF	0.780	0.770
TF-IDF	0.740	0.760

This table reports the accuracy scores for the MLP model using textual features as inputs. Particularly, we use the MLP model with four different word embedding features: (1) TF Centroid with generic word embeddings as inputs (TF Generic centroid), (2) TF-IDF Centroid with generic word embeddings as inputs (TF-IDF Generic centroid), (3) TF Centroid with finance word embeddings as inputs (TF Finance centroid), and (4) TF-IDF Centroid with finance word embeddings as inputs (TF-IDF Finance centroid). The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table 6

Out-of-sample performance using both SVD100 textual features based on the bag of word approach and financial variables as inputs

	SVM-linear	SVM-RBF	LOGIT	RF	MLP
Panel A: Bidders					
TF_{SVD100}	0.535	0.522	0.643	0.618	0.651
$TF\text{-}IDF_{SVD100}$	0.537	0.563	0.628	0.672	0.641
(TF + bigrams)$_{SVD100}$	0.450	0.545	0.641	0.643	0.651
(TF-IDF + bigrams)$_{SVD100}$	0.509	0.579	0.628	0.651	0.623
Panel B: Targets					
TF_{SVD100}	0.699	0.647	0.724	0.808	0.776
$TF\text{-}IDF_{SVD100}$	0.615	0.660	0.744	0.853	0.801
(TF + bigrams)$_{SVD100}$	0.647	0.673	0.679	0.814	0.756
(TF-IDF + bigrams)$_{SVD100}$	0.615	0.647	0.718	0.897	0.807

This table reports the accuracy scores for our machine learning models, using both textual features based on the bag of word approach and financial variables. The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. However, the dimensions of textual features are further reduced to 100 using the singular value decomposition dimensionality reduction technique (SVD100). We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. The first two lines of each panel report results using only unigrams, while the last two lines report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature. We use the following machine learning models: support vector machines (SVM-linear), support vector machines with radial basis function kernel (SVM-RBF), logistic regression (LOGIT), random forest (RF), and multilayer perceptron (MLP). TF and TF-IDF represent the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table 7

Out-of-sample performance of the MLP model using both textual features based on the word embedding approach and financial variables as inputs

	Generic centroid	Finance centroid
Panel A: Bidders		
TF	0.670	0.690
TF-IDF	0.680	0.720
Panel B: Targets		
TF	0.790	0.800
TF-IDF	0.780	0.810

This table reports the accuracy scores for the MLP model using both textual features and financial variables as inputs. Particularly, we use the MLP model with four different word embedding features: (1) TF Centroid with generic word embeddings as inputs (TF Generic centroid), (2) TF-IDF Centroid with generic word embeddings as inputs (TF-IDF Generic centroid), (3) TF Centroid with finance word embeddings as inputs (TF Finance centroid), and (4) TF-IDF Centroid with finance word embeddings as inputs (TF-IDF Finance centroid). The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. TF and TF-IDF represent the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table 8

Bootstrap randomization and statistical significance

Comparisons	Winner	p-value
Panel A: Bidders		
TF-IDF Finance centroid *vs* TF-IDF Generic centroid	TF-IDF Finance centroid	0.006***
TF-IDF Finance centroid *vs* TF Finance centroid	No winner	0.118
TF-IDF Finance centroid *vs* TF Generic centroid	TF-IDF Finance centroid	0.000***
TF-IDF Generic centroid *vs* TF Finance centroid	No winner	0.452
TF-IDF Generic centroid *vs* TF Generic centroid	TF-IDF Generic centroid	0.049**
TF Finance centroid *vs* TF Generic centroid	TF Finance centroid	0.000***
Panel B: Targets		
RF-(TF-IDF + bigrams)$_{SVD100}$ *vs* RF-TF-IDF$_{SVD100}$	RF-(TF-IDF + bigrams)$_{SVD100}$	0.000***
RF-(TF-IDF + bigrams)$_{SVD100}$ RF *vs* RF-(TF + bigrams)$_{SVD100}$	RF-(TF-IDF + bigrams)$_{SVD100}$	0.000***
RF-(TF-IDF + bigrams)$_{SVD100}$ RF *vs* RF-TF$_{SVD100}$	RF-(TF-IDF + bigrams)$_{SVD100}$	0.000***
RF-TF-IDF$_{SVD100}$ *vs* RF-(TF + bigrams)$_{SVD100}$	RF-TF-IDF$_{SVD100}$	0.036**
RF-TF-IDF$_{SVD100}$ *vs* RF-TF$_{SVD100}$	RF-TF-IDF$_{SVD100}$	0.014**
RF-(TF + bigrams)$_{SVD100}$ *vs* RF-TF$_{SVD100}$	No winner	0.684

This table reports the p-values of our results based on bootstrap statistical significance tests. In each task (bidders or targets), we choose the four best-performing models. In practice, we compare the chosen models' performance in order to classify them in terms of predictive power, however with respect to statistical significance. Panel A reports results when we attempt to predict bidders. Particularly, we use the MLP model with four different word embedding features: (1) TF Centroid with generic word embeddings as inputs (TF Generic centroid), (2) TF-IDF Centroid with generic word embeddings as inputs (TF-IDF Generic centroid), (3) TF Centroid with finance word embeddings as inputs (TF Finance centroid), and (4) TF-IDF Centroid with finance word embeddings as inputs (TF-IDF Finance centroid). Panel B reports results when we attempt to predict targets. In particular, we compare the following four models: (1) RF with the use of TFSVD100 (RF-TF$_{SVD100}$), (2) RF with the use of TF-IDF$_{SVD100}$ (RF-TF-IDF$_{SVD100}$), (3) RF with the use of (TF + bigrams)$_{SVD100}$ (RF-(TF + bigrams)$_{SVD100}$), and (4) RF with the use of (TF-IDF + bigrams)$_{SVD100}$ (RF-(TF-IDF + bigrams)$_{SVD100}$). TF and TF-IDF represent the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Appendix A

Table A1

Financial variables definition

Variables	Description	Commercial Banks (Call Reports)	Bank Holding Companies (FR Y-9C)
Size	Logarithm of Total Assets	ln(RCFD2170)	ln(BHCK2170)
Capital strength	Equity to Total Assets	RCFD3210/RCFD2170	BHCK3210/BHCK2170
Loans	Loans to Total Assets	RCFD2122/RCFD2170	BHCK2122/BHCK2170
Non-interest income	Non-Interest Income to Total Income	RIAD4079/(RIAD4074+RIAD4079)	BHCK4079/(BHCK4074+BHCK4079)
Asset quality	Loan Loss Provisions to Loans	RIAD4230/RCFD2122	BHCK4230/BHCK2122
Deposits	Deposits to Total Assets	(RCFD6631+RCFD6636)/RCFD2170	(BHDM6631+BHDM6636+BHFN6631+BHFN6636)/BHCK2170
Market power	Deposits market share	(RCFD6631+RCFD6636)/Total industry deposits	(BHDM6631+BHDM6636+BHFN6631+BHFN6636)/Total industry deposits
Cost efficiency	Non-Interest Expense to Total Income	RIAD4093/(RIAD4074+RIAD4079)	BHCK4093/(BHCK4074+BHCK4079)
ROA	Net Income to Total Assets	RIAD4340/RCFD2170	BHCK4340/BHCK2170

This table presents the construction of our financial variables and corresponding codes from the Call Reports and FR Y-9C Reports.

Table A2

Out-of-sample performance using both textual features based on the bag of words approach and financial variables as inputs

	SVM-linear	SVM-RBF	LOGIT	RF	MLP
Panel A: Bidders					
TF	0.475	0.506	0.607	0.643	0.646
TF-IDF	0.527	0.501	0.615	0.633	0.612
TF + bigrams	0.504	0.506	0.615	0.680	0.662
TF-IDF + bigrams	0.506	0.501	0.576	0.625	0.636
Panel B: Targets					
TF	0.596	0.519	0.801	0.673	0.808
TF-IDF	0.603	0.532	0.750	0.769	0.833
TF + bigrams	0.545	0.538	0.750	0.744	0.801
TF-IDF + bigrams	0.519	0.526	0.769	0.795	0.840

This table reports the accuracy scores for our machine learning models, using both textual features based on the bag of words approach and financial variables. The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. The first two lines of each panel report results using only unigrams, while the last two lines report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature. We use the following machine learning models: support vector machines (SVM-linear), support vector machines with radial basis function kernel (SVM-RBF), logistic regression (LOGIT), random forest (RF), and multilayer perceptron (MLP). TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table A3

Out-of-sample performance using only SVD100 textual features based on the bag of words approach

	SVM-linear	SVM-RBF	LOGIT	RF	MLP
Panel A: Bidders					
TF_{SVD100}	0.553	0.476	0.631	0.581	0.612
$TF\text{-}IDF_{SVD100}$	0.507	0.519	0.628	0.631	0.649
$(TF + bigrams)_{SVD100}$	0.630	0.463	0.646	0.605	0.631
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.519	0.563	0.641	0.625	0.641
Panel B: Targets					
TF_{SVD100}	0.596	0.564	0.750	0.756	0.737
$TF\text{-}IDF_{SVD100}$	0.628	0.680	0.763	0.795	0.776
$(TF + bigrams)_{SVD100}$	0.590	0.545	0.756	0.743	0.731
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.609	0.667	0.769	0.801	0.795

This table reports the accuracy scores for our machine learning models, using only textual features based on the bag of words approach. The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. However, the dimensions of textual features are further reduced to 100 using the singular value decomposition dimensionality reduction technique (SVD100). We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. The first two lines of each panel report results using only unigrams, while the last two lines report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature. We use the following machine learning models: support vector machines (SVM-linear), support vector machines with radial basis function kernel (SVM-RBF), logistic regression (LOGIT), random forest (RF), and multilayer perceptron (MLP). TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table A4

Out-of-sample performance using both SVD100 textual features and financial variables as inputs after excluding special years

	SVM-linear	SVM-RBF	LOGIT	RF	MLP
Panel A: Bidders					
TF_{SVD100}	0.578	0.523	0.652	0.635	0.641
$TF\text{-}IDF_{SVD100}$	0.543	0.572	0.627	0.667	0.655
$(TF + bigrams)_{SVD100}$	0.586	0.526	0.667	0.658	0.647
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.552	0.578	0.626	0.670	0.658
Panel B: Targets					
TF_{SVD100}	0.591	0.644	0.700	0.818	0.727
$TF\text{-}IDF_{SVD100}$	0.492	0.674	0.742	0.845	0.765
$(TF + bigrams)_{SVD100}$	0.621	0.598	0.689	0.841	0.795
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.553	0.750	0.720	0.871	0.811

This table reports the accuracy scores for our machine learning models, using both textual features based on the bag of words approach and financial variables. The final sample consists of 7,456 bank-year observations from 1994 to 2016, after we remove all bank-year observations from years 2000-2001 as the years of dot-com bubble, and from years 2008-2009 as the years of the financial crisis. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. However, the dimensions of textual features are further reduced to 100 using the singular value decomposition dimensionality reduction technique (SVD100). We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 868 bidders and 868 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 328 targets and 328 non-involved banks. The first two lines of each panel report results using only unigrams, while the last two lines report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature. We use the following machine learning models: support vector machines (SVM-linear), support vector machines with radial basis function kernel (SVM-RBF), logistic regression (LOGIT), random forest (RF), and multilayer perceptron (MLP). TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table A5

Out-of-sample performance of the MLP model using both textual features based on the word embedding approach and financial variables as inputs after excluding special years

	Generic centroid	Finance centroid
Panel A: Bidders		
TF	0.670	0.690
TF-IDF	0.680	0.710
Panel B: Targets		
TF	0.790	0.800
TF-IDF	0.780	0.800

This table reports the accuracy scores for the MLP model using both textual features and financial variables as inputs. Particularly, we use the MLP model with four different word embedding features: (1) TF Centroid with generic word embeddings as inputs (TF Generic centroid), (2) TF-IDF Centroid with generic word embeddings as inputs (TF-IDF Generic centroid), (3) TF Centroid with finance word embeddings as inputs (TF Finance centroid), and (4) TF-IDF Centroid with finance word embeddings as inputs (TF-IDF Finance centroid). The final sample consists of 7,456 bank-year observations from 1994 to 2016, after we remove all bank-year observations from years 2000-2001 as the years of dot-com bubble, and from years 2008-2009 as the years of the financial crisis. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 868 bidders and 868 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 328 targets and 328 non-involved banks. TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table A6

Out-of-sample performance of the 10,000 most frequent textual features using both SVD100 textual features and financial variables as inputs

	SVM-linear	SVM-RBF	LOGIT	RF	MLP
Panel A: Bidders					
TF_{SVD100}	0.535	0.535	0.643	0.631	0.627
$TF\text{-}IDF_{SVD100}$	0.494	0.574	0.628	0.666	0.636
$(TF + bigrams)_{SVD100}$	0.434	0.532	0.633	0.643	0.635
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.574	0.587	0.625	0.659	0.638
Panel B: Targets					
TF_{SVD100}	0.603	0.699	0.724	0.814	0.756
$TF\text{-}IDF_{SVD100}$	0.571	0.705	0.744	0.859	0.802
$(TF + bigrams)_{SVD100}$	0.635	0.686	0.699	0.846	0.776
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.692	0.602	0.724	0.865	0.782

This table reports the accuracy scores for our machine learning models, using both textual features based on the bag of words approach and financial variables. The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 10,000 most frequent words of the 10-K filing. However, the dimensions of textual features are further reduced to 100 using the singular value decomposition dimensionality reduction technique (SVD100). We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. The first two lines of each panel report results using only unigrams, while the last two lines report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature. We use the following machine learning models: support vector machines (SVM-linear), support vector machines with radial basis function kernel (SVM-RBF), logistic regression (LOGIT), random forest (RF), and multilayer perceptron (MLP). TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Table A7
Sum of Gini impurity scores

	Financial variables Gini	Textual variables Gini
Panel A: Bidders		
TF_{SVD100}	0.075	0.169
$TF\text{-}IDF_{SVD100}$	0.054	0.194
$(TF + bigrams)_{SVD100}$	0.092	0.152
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.054	0.117
Panel B: Targets		
TF_{SVD100}	0.167	0.172
$TF\text{-}IDF_{SVD100}$	0.111	0.243
$(TF + bigrams)_{SVD100}$	0.141	0.189
$(TF\text{-}IDF + bigrams)_{SVD100}$	0.113	0.222

This table reports the Gini impurity scores when both SVD100 textual features and financial variables are used as inputs in the RF model. In fact, we provide the sum of Gini scores separately for financial variables and textual features for comparative reasons. However, in our calculations we take into account only the 20 most important features. The final (imbalanced) sample consists of 9,207 bank-year observations from 1994 to 2016. To construct the textual features, we use the 20,000 most frequent words of the 10-K filing. However, the dimensions of textual features are further reduced to 100 using the singular value decomposition dimensionality reduction technique (SVD100). We use 80% of our sample as the training set and the remaining 20% as the out-of-sample (testing set). Panel A reports results when we attempt to predict bidders. The analysis is based on a balanced sample of 966 bidders and 966 non-involved banks. Panel B reports results when we attempt to predict targets. The analysis is based on a balanced sample of 389 targets and 389 non-involved banks. The first two lines of each panel report results using only unigrams, while the last two lines report results using combinations of unigrams and bigrams. Bigrams are pairs of consecutive words represented as a single textual feature. TF and TF-IDF are the two term weighting schemes. TF stands for the term frequency scheme normalized by document length, and TF-IDF for the term frequency-inverse document frequency scheme.

Appendix B

In this section, we describe the bootstrap resampling method of Berg-Kirkpatrick et al. (2012). To understand this approach, let's assume that we want to compare the out-of-sample performance of two learning algorithms, algorithm A and algorithm B. We construct the null hypothesis H_0, which assumes that A is no better than B when it comes to their performance. If we fail to reject H_0, then, any outperformance of A could be attributed to chance. We test this hypothesis by estimating the p-value of the null hypothesis using the bootstrap method, which simulates several out-of-samples (testing sets) from the original out-of-sample.[18]

In particular, let's suppose that a classifier A is better than another classifier B by δ(x) based on a test set $x = x_1, \ldots, x_n$. We then generate b versions of out-of-sample x using the bootstrap method with replacement. To estimate how unexpected our observed δ(x) is, we compute the following p-value:

$$P(\delta(X) > \delta(x)|H_0)$$

where X is a random variable over potential out-of-samples of size n drawn from the bootstrap method, and $\delta(x)$ is a constant that reflects the observed performance advantage of A over B. In essence, we check how frequently classifier A beats B by a more than $\delta(x)$ accuracy score on $x^{(i)}$, where i takes values from 1 to 10,000 and represents the new out-of-samples, known as bootstrap samples. Nevertheless, the $x^{(i)}$ were sampled from x, which implies that the average $\delta\left(x^{(i)}\right)$ would undoubtedly be not zero, as we expressed in the null hypothesis. In other words, the bootstrap samples are drawn from x, which is biased in favor of A by the amount of $\delta(x)$. The expected value of $\delta(X)$ tends to be close to δ(x), instead of zero. To deal with that, we re-center the expected

[18] In practice, we pool the test sets from the underlying classifiers. Next, we create new test sets, called bootstrap samples, and drawn from the pooled sample with replacement. For the purposes of our analysis, we repeat the experiment 10,000 times.

value, by counting the number of $x^{(i)}$ where A beating B with at least $2 * \delta(x)$. The p-value is computed as follows:

$$P(\delta(X) - \delta(x) > \delta(x)|H_0)$$

$$P(\delta(X) > 2 * \delta(x)|H_0)$$