Next-Year Bankruptcy Prediction from Textual Data: Benchmark and Baselines

Henri Arno[1*] and **Klaas Mulier**[1] and **Joke Baeck**[1] and **Thomas Demeester**[2]
[1]Ghent University
[2]Ghent University - imec
first.last@UGent.be

Abstract

Models for bankruptcy prediction are useful in several real-world scenarios, and multiple research contributions have been devoted to the task, based on structured (numerical) as well as unstructured (textual) data. However, the lack of a common benchmark dataset and evaluation strategy impedes the objective comparison between models. This paper introduces such a benchmark for the unstructured data scenario, based on novel and established datasets, in order to stimulate further research into the task. We describe and evaluate several classical and neural baseline models, and discuss benefits and flaws of different strategies. In particular, we find that a lightweight bag-of-words model based on static in-domain word representations obtains surprisingly good results, especially when taking textual data from several years into account. These results are critically assessed, and discussed in light of particular aspects of the data and the task. All code to replicate the data and experimental results will be released.

1 Introduction

Since the seminal work of Beaver (1966), bankruptcy prediction has received considerable attention by both academics and practitioners. A sound prediction model has numerous applications. For instance, successful quantitative methods can help professionals, such as creditors and investors, in managing financial risk (Bielecki and Rutkowski, 2013). Furthermore, as Bernanke (1981) has shown that economy-wide levels of bankruptcy risk play a structural role in propagating recession, regulators can use bankruptcy prediction models to monitor the financial health of key economic actors and control systematic risk.

A large number of bankruptcy prediction models have been proposed in literature, such as the models from Beaver (1966), Ohlson (1980), Odom and Sharda (1990), Kim and Kang (2010) and Mai et al. (2019). However, it appears difficult to compare these studies and objectively assess progress in the field. We have identified the following three aspects that make comparison difficult: (1) the temporal nature and typical class imbalance of the bankruptcy prediction task leads to strongly deviating evaluation scenarios, (2) there is little consensus on the key evaluation metrics, and (3) there is no standard benchmark dataset. These issues are further discussed in section 2.2. In order to overcome these problems, we have designed and described our experimental setup with reproducibility on a common benchmark in mind. To that end, scripts to reconstruct the benchmark and reproduce the presented results are available at https://github.com/henriarnoUG/BankruptcyBenchmarkBaselines. Note that this paper investigates the potential to predict bankruptcy from textual disclosures only. Extending this benchmark to the hybrid case of combined textual and structured features will be part of our future work.

The contributions of this paper are as follows: (1) we introduce a reproducible benchmark for text-based bankruptcy prediction, based on novel and established economic datasets, (2) classical as well as neural baseline prediction models are provided, including results on next-year bankruptcy prediction from multiple years of textual data, and (3) insights into the results are given along with pointers to potential next steps in bankruptcy prediction.

2 Related Work

After a general overview of research on bankruptcy prediction (Section 2.1), we describe some key aspects that make contributions in literature hard to compare (Section 2.2).

* Corresponding author

Proceedings of the Fourth Workshop on Financial Technology and Natural Language Processing (FinNLP), pages 187 - 195

Three years prior to bankruptcy	*"We are highly leveraged and a substantial portion of our liquidity needs arise from debt service requirements and from funding our costs of operations and capital expenditures, including acquisitions... we entered into a new asset-based revolving credit facility (ABL Facility)... secured by substantially all of our assets..."*
One year prior to bankruptcy	*" ... we received a waiver of certain events of default under the TLA arising from the inclusion of a going concern qualification from our registered public accounting firm, breach of the EBITDA financial covenant, and cross-default arising from the default under our ABL Facility... In order to address our liquidity issues and provide for a restructuring of our indebtedness to improve our long-term capital structure, we have entered into a Restructuring Support Agreement ... pursuant to a prepackaged plan of reorganization to be filed in a case commenced under chapter 11 of the United States Bankruptcy Code..."*

Table 1: Extracts from the MD&A section of a distressed company in our dataset, one year and three years prior to bankruptcy. Underlined words correspond to the top 20 tokens most informative for imminent bankruptcy in our respective Binary Bag-of-Words models.

2.1 Bankruptcy Prediction Research

Beaver (1966) pioneered bankruptcy prediction literature with a discriminant model based on financial ratios. Subsequently, well-chosen structured financial variables were proposed to predict failure, along with increasingly advanced prediction models. Statistical models, such as discriminant analysis (Beaver, 1966; Altman, 1968), have been dominant in the past but rely on stringent assumptions about the data (Balcaen and Ooghe, 2006). Today, machine learning models are commonplace as they rely on fewer assumptions and learn directly from the data. Odom and Sharda (1990) used neural networks to predict bankruptcy, Kim and Kang (2010) have built an ensemble model and Hosaka (2019) generates predictions through a convolutional neural network with ratios presented as images. Keasey and Watson (1987) were the first to include non-financial variables in a corporate failure model, Shumway (2001) has shown that market-driven variables are strongly related to bankruptcy and Cecchini et al. (2010) found that textual disclosures can be used to discriminate between bankrupt and non-bankrupt firms. The information value of textual data was further established by Mayew et al. (2015) as they found that the opinion of management on the future of the company and the linguistic tone of the Management Discussion and Analysis has significant explanatory power for corporate failure. Mai et al. (2019) provide large-sample evidence of the predictive power of textual disclosures and show that deep learning models yield superior results when using textual data together with traditional accounting features. Furthermore, the authors compare two deep learning architectures based on skip-gram word representations (Mikolov et al., 2013) and conclude that an average embedding model leads to better results than a ConvNet architecture. Despite this promising work, bankruptcy prediction models using textual data are scarce.

2.2 Need for a Reproducible Benchmark

The following aspects prevent a straightforward comparison of research contributions, and may be avoided by a common benchmark along with the tools to reproduce experimental results, one of the goals of this work.

Temporal nature and class imbalance of bankruptcy data: Due to the temporal nature of the data and the typically much smaller fraction of positive cases (enterprises going bankrupt), many strategies have been proposed to construct training data and define evaluation sets. The data source that serves as a basis for the model typically contains annual (or more fine-grained) observations for each firm in the sampling period. In earlier work (Beaver, 1966; Altman, 1968) the explanatory variables were selected only once for each firm in the dataset. In the 'paired sampling' approach (Altman, 1968), the independent variables for failed firms were retained in the year before failure, together with those for a paired healthy firm in that same year, to induce a balanced dataset from which a random evaluation set is sampled. Shumway (2001) has shown that such an approach leads to poor out-of-sample prediction performance and incorrect statistical inference. As an alternative, hazard models can be estimated by treating each firm-year sample as an independent observation, with the bankruptcy status by the end of the following year as the prediction target. Typically, the observations prior to some date are used for model training, and

observations after this date are used to estimate the out-of-period prediction performance (Shumway, 2001; Mai et al., 2019). Sometimes even a random split is used, independent of time (Mai et al., 2019). In the work of Volkov et al. (2017), the explanatory variables for a number of consecutive years are used as input, with company status as the prediction target in the year afterwards. The class imbalance is managed through undersampling of healthy companies. Evaluation is done on a held-out subset of companies, which is therefore artificially balanced as well. Undersampling, oversampling, and data augmentation techniques are investigated by Veganzones and Séverin (2018). Training and evaluation are done on a non-overlapping subset of firms, with a one-year shift in between, while also maintaining a predefined artificial ratio between the number of healthy and bankrupt firms (for both training and evaluation).

In our considered population (public companies in the US, see Section 3.1), all companies are known, as well as their yearly reports so far, and the goal is predicting bankruptcy for all of these firms in the near future (the coming year). This is simulated in our evaluation scenario, where we make predictions for *all* companies not (yet) bankrupt and observed through annual reports up to a given year, on their bankruptcy status the year afterwards (as further detailed in section 3.2).

Large variety of evaluation metrics: The choice of evaluation metrics is often linked to the experimental setup, e.g., depending on whether a balanced test set is used. The evaluation scenario also influences the choice of threshold used for metrics like accuracy, precision, or recall. For example, Volkov et al. (2017) select a threshold that maximises the F_2-measure. Alternatively, Veganzones and Séverin (2018) select the threshold that minimises the expected cost of misclassification with equal weights. Aggregated metrics that avoid the use of a threshold, such as area under the ROC curve (AUC), decile rank, and cumulative accuracy profile ratio (CAP) are regularly reported as well (Mai et al., 2019).

Use of private datasets: The final reason that makes model comparison hard is the lack of a standard benchmark dataset. Bankruptcy prediction literature either reports results on proprietary datasets (Matin et al., 2019) or on data obtained by manual collection or custom web scraping strategies (and kept private) (Cecchini et al., 2010; Wang et al., 2020). For a comprehensive overview of data sources used in recent corporate failure literature we refer the reader to the work of Mai et al. (2019). Our datasets are based on the combination of existing sources, i.e., the UCLA-LoPucki Bankruptcy Research Database (BRD)[1] and the public EDGAR-CORPUS (Loukas et al., 2021). This allows researchers to reconstruct the same train, validation and test data from these sources, even if we are not allowed to make the resulting datasets public directly.

3 Methodology

In the next sections, we describe the data sources (Section 3.1) and motivate our design choices for the benchmark (Section 3.2), document preprocessing (Section 3.3), and the selected evaluation metrics (Section 3.4).

3.1 Data Sources

Our study makes use of the EDGAR-CORPUS, a novel economic dataset containing 10-k reports from all publicly traded companies in the US, spanning 25 years (Loukas et al., 2021). As we need information on bankruptcies as prediction target, these reports were matched with the UCLA-LoPucki Bankruptcy Research Database (the BRD)[2], through the unique Central Index Key to identify companies. The BRD contains information on all Chapter 7 and Chapter 11 filings of the United States Bankruptcy Code since 1997 and is updated monthly.

Consistent with prior work (Cecchini et al., 2010; Mayew et al., 2015; Mai et al., 2019), we limit the 10-k reports to section 7: "Management Discussion and Analysis". According to the U.S. Securities and Exchange Commission[3], it *"... gives the company's perspective on the business results of the past financial year. This section, known as the MD&A for short, allows company management to tell its story in its own words."* It also contains the risks and uncertainties that could materially affect the company. As an example, consider the extracts from the MD&A's of a distressed firm in Table 1.

Public company bankruptcy is a rare event. Figure 1 shows that the number of 10-k reports filed by

[1] https://lopucki.law.ucla.edu/

[2] The BRD does require a paid annual subscription or a one-time purchase for academic single use.

[3] https://www.sec.gov/fast-answers/answersreada10khtm.html

non-bankrupt companies heavily exceeds the yearly number of Chapter 7 and Chapter 11 cases. Note how the influence of the Dot-com crisis (2000), the financial crisis (2007-2008), and the COVID crisis (2020) on our population can be observed. Table 2 provides additional statistics for the aligned data sources.

3.2 Task Definition and Setup

3.2.1 Determining the prediction time window

Prior work has not always been very transparent about the temporal aspect of the textual and numerical data in their models, but this requires special attention in order to arrive at a correct setup. A 10-k report is characterised by two dates, as schematically shown in Fig. 2: (1) the fiscal year-end t_{PR} of the one-year time window T_{PR} ('period of report') used to calculate the financial statements, and (2) the filing date t_{FD} on which the report is filed with the SEC. Since in practice $t_{FD} \geq t_{PR}$, there may be a period after t_{PR} yielding textual information in the MD&A (i.e., before t_{FD}), not present in the financial statements. It is therefore important to use the one-year period directly *after* t_{FD} as the prediction time window $T_{prediction}$ when the textual data is used as input to the model. In the extreme case of bankruptcy in between t_{PR} and t_{FD} ('potential bankruptcy' in Fig. 2), it would lead to leakage and artificially high prediction accuracies if the year directly after t_{PR} were used for prediction. It is possible, though, that information on an imminent bankruptcy shortly *after* t_{FD} is already included in the report, but this does not present a conceptual problem for the prediction setup.

3.2.2 Dealing with missing 10-k reports

The dataset contains yearly 10-k reports from the first time a company appears, starting from the year 2000, until 2021 or until bankruptcy. However, some reports are missing for a number of companies, and our analysis reveals the following three scenarios. First, some companies stop reporting from a certain point in time onwards, without filing for bankruptcy. This may be due to a merger or an acquisition, but that particular information is not present in the data. Second, there may be gaps in the sequence of yearly reports. This arises when a company either does not submit a 10-k report (due to unknown reasons) or because of data quality issues. Third, we observe that some companies headed towards bankruptcy tend to fail in their reporting in the year(s) leading up to the bankruptcy

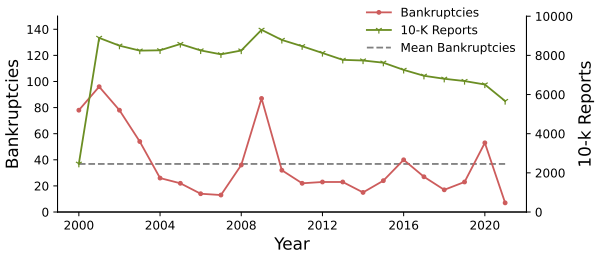

Figure 1: The number of bankruptcies (including the mean) (left y-axis) and the number of 10-k reports filed (right y-axis) per year.

period	2000-2021
avg. reports per year	7599 ± 1477
avg. bankruptcies per year	39 ± 26
avg. new enterprises per year	1467 ± 1376
avg. doc. length (# tokens)	6492 ± 1138

Table 2: Summary statistics of our aligned data sources.

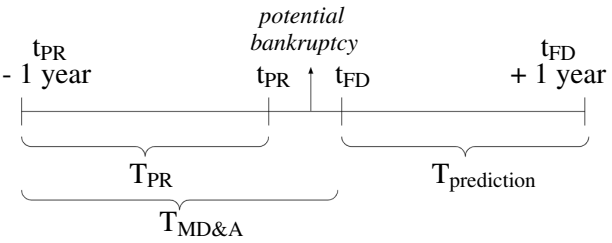

Figure 2: Timeline containing the characterising dates (t_{PR}, t_{FD}) of a 10-k report and corresponding periods (T_{PR}, $T_{MD\&A}$, $T_{prediction}$)

filing. A naive approach would be to simply discard all instances with missing reports. However, this would make the evaluation scenario biased, since missing reports are not distributed uniformly over the data, due to the different scenarios described above.

Consider our 2019 test set with a history of three years (discussed later in this section) as an example, of which close to 45% of companies have at least one missing report during the three-year history. The relative frequency of bankruptcy is 0.27% for the entire population, 0.00% for companies with only missing data (cf. an M&A event), 0.35% for companies with no missing data and 0.93% for companies where the data in only the year before prediction is missing. Therefore, we do not remove these companies and keep them in our dataset which results in a more realistic evaluation scenario.

3.2.3 Construction of input and target per firm-year

In order to create time-agnostic firm-year samples (following Shumway (2001)) during the construction of our train, validation and test sets (see further), we process a given year and company as follows:

1. **Determine $T_{prediction}$:** If a 10-k report was filed by the company in the considered year, $T_{prediction}$ is the period between t_{FD} and t_{FD} + 1 year (cf. Figure 2). Otherwise, we use the one-year period starting the same day as the latest available t_{FD}, but in the considered year.

2. **Assign target label:** If the company filed for bankruptcy during $T_{prediction}$, the label is 1, otherwise 0. Note that potential firm-year instances with a bankruptcy filing *before* t_{FD} are invalid for the considered year, as explained above.

3. **Collect textual data:** The MD&A text from the report filed at t_{FD} is used for the one-year history setting, as well as from the two previous years for the three-year scenario. For missing reports, the token 'missing' is used.

3.2.4 Train / validation / test segmentation

Training data: We construct two training sets in total. The first, using data up to 2015, is used for initial training while leaving sufficient data for validation during hyperparameter tuning. The second, with data up to 2017, is used to train the final models. They are constructed as follows:

1. We leave out all reports with a t_{FD} later than 2015 (2017), to ensure a proper temporal split between training and evaluation data.

2. For every firm and every year between the first year of the training data and 2015 (2017), we construct a firm-year instance as described above.

3. To reduce the impact on the training process of instances without any reports in their considered history (i.e., the one-year or three-year history, respectively), 95% of those are randomly removed.

Validation data: We construct two validation sets, one for 2017 and one for 2018, both to be used for hyperparameter tuning. First, we filter out companies that have not filed any reports during the 5 years leading up to and including 2017 (2018). For each of these companies, one firm-year sample is created according to the method described above for the year (and hence t_{FD}, even when the report is missing) 2017 (2018).

Test data: In the same way, we construct two test sets, one for 2019 and one for 2020 (denoting the calendar year containing t_{FD}), for the final evaluation of the trained models.

3.3 Pre-processing

When dealing with textual data it is common to perform document pre-processing in order to decrease the dimensionality of the problem and reduce the computational cost of encoding the documents. We perform four pre-processing steps for the Bag-of-Words models presented in sections 4.1-4.3. First, we lowercase all documents. Second, we remove stopwords and punctuation. Third, we lemmatize each word in the documents through the NLTK library (Loper and Bird, 2002). Inflicted word forms such as *paying* and *payed* are transformed into the root form *pay*. Finally, we replace uncommon words by the token '_UNK_' (for 'unknown'). A word is deemed uncommon when it does not appear in the 50,000 most frequent words in the training set. When dealing with transformer models (Vaswani et al., 2017), such as the Longformer (Beltagy et al., 2020), these steps are typically not required and might even lead to deteriorating performance. Preprocessing then consists of proper tokenization of the input text. We use the tokenization tools from Huggingface [4], which allow transforming the input text into a sequence of well-chosen word pieces.

3.4 Evaluation Metrics

Following Mai et al. (2019), we report the **Area Under the Receiver Operating Curve** (AUC) as main evaluation metric. The AUC is often used to quantify the overall prediction performance of binary decision models. It aggregates the information in the Receiver Operator Curve (ROC), which quantifies the trade-off between the true positive rate (or recall) and the false positive rate at various classification thresholds. However, in certain scenarios, a high true positive rate may be more relevant than a low false positive rate. Therefore, we

[4] https://huggingface.co/

	Single year history				Three year history			
	Binary	TF-IDF	W2V	Longformer	Binary	TF-IDF	W2V	Longformer
AUC	0.79 (0.84)	0.80 (0.85)	**0.88 (0.90)**	0.78 (0.79)	0.90 (0.92)	0.92 (0.96)	**0.95 (0.95)**	0.85 (0.84)
AP	0.07 (0.05)	**0.16 (0.16)**	0.08 (0.12)	0.01 (0.03)	0.03 (0.06)	**0.10 (0.10)**	0.04 (0.09)	0.02 (0.02)
rec@100	0.19 (0.18)	0.26 (0.31)	**0.37 (0.31)**	0.04 (0.07)	0.15 (0.22)	**0.37 (0.29)**	0.22 (0.24)	0.11 (0.02)
CAP	0.56 (0.68)	0.59 (0.72)	**0.75 (0.80)**	0.52 (0.58)	0.82 (0.84)	0.83 (0.92)	**0.89 (0.89)**	0.71 (0.68)
1	0.56 (0.67)	**0.74 (0.71)**	0.70 (0.73)	0.56 (0.51)	0.78 (0.73)	0.70 (0.91)	**0.85 (0.84)**	0.41 (0.40)
2	0.74 (0.78)	0.74 (0.84)	**0.78 (0.80)**	0.70 (0.71)	0.89 (0.87)	**0.93 (1)**	0.93 (0.93)	0.78 (0.80)
3	0.78 (0.84)	0.78 (0.87)	**0.85 (0.80)**	0.74 (0.76)	0.96 (0.93)	0.96 (1)	**1 (0.98)**	0.93 (0.91)
4	0.78 (0.87)	0.78 (0.87)	**0.96 (0.91)**	0.74 (0.82)	0.96 (1)	0.96 (1)	**1 (1)**	0.93 (0.93)
5	0.78 (0.87)	0.78 (0.87)	**0.96 (0.98)**	0.74 (0.87)	0.96 (1)	0.96 (1)	**1 (1)**	0.93 (0.96)

Table 3: Bankruptcy prediction results on the test sets: 2019 (2020), for several bag-of-words models: with binary one-hot vectors (Binary), TF-IDF, and mean word-to-vec (W2V) representations, as well as a Longformer classifier, and for single-year vs. three-year text inputs. Reported metrics are the area-under-the-ROC-curve (AUC), average precision (AP), recall@100 (rec@100), cumul. accuracy profile ratio (CAP), and cumul. decile rank (1-5).

also report the **Recall@100**. It quantifies the proportion of positive cases (bankrupt firms) present in the 100 highest ranked ones, out of all positive samples (all bankrupt firms in the considered year). In our context, this metric evaluates the models in their effectiveness to detect as many distressed enterprises as possible for a given budget (e.g., the manpower to investigate a hundred firms). The **Cumulative Accuracy Profile Ratio** (CAP) is a ranking based metric with a strong emphasis on recall of the positive class. It summarises the information in the CAP curve, which plots the cumulative proportion of positive samples against the percentage of the ranked data taken into account. The **Cumulative Decile Rank** is also a recall oriented metric. It gives the cumulative proportion of all positive samples (bankrupt firms) in each decile when ranking the samples according to the classifier score. Although we consider recall more important for the bankruptcy case from the perspective of the 'given budget' scenario outlined above, we report a precision oriented metric as well. The **Average Precision** (AP) is the weighted mean of the precision at each classification threshold with the increase in recall as weight.

4 Models

Sections 4.1-4.3 introduce our bag-of-words (BoW) models (which discard word order), followed by a neural sequence encoder model that does account for word order (Section 4.4), and some training details (Section 4.5).

4.1 Binary Bag-of-Words Model

As a trivial baseline (referred to as 'Binary') we represent our documents as vocabulary-sized binary vectors with '1' at a particular position indicating the presence of the corresponding word. As vocabulary, all occurring unigrams and bigrams are initially considered as features, and reduced to the 20 most informative ones through univariate feature selection, to be used in a logistic regression classifier. This baseline intends to quantify how well the occurrence of a small set of keywords allows predicting bankruptcy. The model for three-year history is obtained the same way, from the joint BoW over the considered years.

4.2 TF-IDF Bag-of-Words Model

The second model is similar to the Binary baseline, but considers *term frequency - inverse document frequency* (TF-IDF) features (Manning et al., 2008) rather than binary ones, combined with feature selection and an L2-regularized logistic regression classifier. The number of features to retain and the inverse regularisation strength are treated as hyperparameters. The three-year model is constructed the same way, after concatenating the texts per year.

4.3 Word2Vec Average Embedding Model

As a final bag-of-words model (W2V), we implement the best performing architecture proposed by Mai et al. (2019), based on the Word2Vec model of Mikolov et al. (2013). First, the pre-processed data is used to train skip-gram word representations of dimension 100 (consistent with Mai et al. (2019)). Documents are then represented by the mean word vector over all occurring words. These serve as input to a two-layer feed-forward neural network with ReLU activations (Glorot et al., 2011) and standard dropout (Srivastava et al., 2014), followed by a sigmoid output. During training, we minimize the binary cross entropy loss with an L2-penalty,

using the Adam optimizer (Kingma and Ba, 2014). The learning rate, weight decay (L2-penalty), hidden layer width, and dropout rate are treated as hyperparameters. When performing classification based on a history of three years, the document representations of each year are concatenated, resulting in a 300-dimensional input to the first hidden layer of the neural network.

4.4 Longformer

For our most advanced neural model, we encode the documents through the Longformer of Beltagy et al. (2020). This transformer-based model is able to handle sequences up to 4096 tokens through its attention mechanism that scales linearly with the input text length (as opposed to the quadratic behavior in earlier Transformer models such as BERT (Devlin et al., 2018). Given the mean document length of over 6k words in our corpus (cf. Table 2), we considered the Longformer a plausible baseline. We process the first 4096 tokens of each document with the Longformer model and retain the 768-dimensional pooled output as the document representation that feeds the same feed-forward classification neural network as described above. For dealing with a history of three years, the individual representations per year are again concatenated, and the input size of the first hidden layer is adjusted accordingly. During training, these representations are kept static (i.e., the Longformer weights are not further fine-tuned on our classification task).

4.5 Training Details

The classical models (Sections 4.1 and 4.2) are implemented in scikit-learn[5] and the hyperparameters are optimised through a grid search procedure. As constructing the vocabulary of all tokens in the training data is expensive, we choose to undersample the majority class until a 90%-10% distribution was reached. The neural models (Sections 4.3 and 4.4) are implemented in PyTorch[5] while the Word2Vec model was trained with Gensim[5] and the forward pass through the Longformer was performed with Huggingface[4]. Since hyperparameter optimisation for deep learning models is expensive, we made use of the Tree-Structured Parzen Estimation algorithm to find the optimal hyperparameter settings (Bergstra et al., 2011) implemented in Optuna[5]. The hyperparameters are tuned to maximise the weighted AUC of the 2017 and 2018 validation data, and the obtained values are then used to train the final models using training data up to 2017, to be tested on the 2019 and 2020 test sets.[6]

Top 15 selected unigrams and bigrams
waiver (0.26), _UNK_ million (0.21), restructuring (0.21), severance (0.20), subordinated (0.20), financial covenant (0.15), indenture (0.14), lender (0.14), interest payment (0.14), senior secured (0.14), asset sale (0.12), senior (0.09), cross default (0.09), indebtedness (0.07), event default (0.05), credit facility (0.05)

Table 4: Top 15 tokens with largest logistic regression coefficients (shown in parentheses) of the Binary bag-of-words model with single year history.

5 Results and Discussion

Table 3 presents the out-of-period test performance metrics for our text-based bankruptcy prediction models, taking a single year or three years of history into account.

When taking a single year of history into account, the W2V model is superior in terms of AUC, recall@100 and CAP while the TF-IDF model achieves the best results in terms of AP. For the 2019 test set, the TF-IDF model contains a slightly higher proportion of positive samples in the first decile but the W2V model is superior from the second decile onwards. When taking three years of history into account, the W2V model achieves the best results for the AUC and CAP metrics while the TF-IDF model performs better with respect to AP and recall@100. When looking at decile rank, the W2V models performs best, having ranked all bankrupt companies in the top 30% of the samples for the 2019 test set.

For each model, AUC and CAP are better when taking three years of history into account compared to a single year of history. The same applies for the decile rank (except for the TF-IDF model and the Longformer model in the first decile). AP is generally worse when using a longer history, except for the Binary model with test set 2020 and the Longformer model with test set 2019. The recall@100 metric varies over the two setups.

We observe that the Binary models based on a mere 20 keywords perform surprisingly well, although not on par with the TF-IDF and W2V models. Note that the latter are based on many more

[5]Scikit-learn: https://scikit-learn.org/stable/
PyTorch: https://pytorch.org/
Optuna: https://optuna.org/
Gensim= https://radimrehurek.com/gensim/

[6]The considered hyperparameter ranges can be accessed through the GitHub repository.

features (in particular, hyperparameter tuning led for the TF-IDF model to 25.000 (10.000) features for single (three) year history). The relatively good performance of the Binary baseline suggests that the presence of few very informative words is a strong indicator for impending bankruptcy. As an illustration, we list the top 15 unigrams and bigrams selected by the single year Binary model in table 4 and underline these features in the extracts in table 1.

Furthermore, the Longformer model performs significantly worse than the other models. Since we do not finetune the generic pre-trained Longformer model on the our end task, the resulting generic document representations appear unable to capture those features in the text that are important for bankruptcy prediction.

The W2V model leads overall to the best results, in particular for AUC (on which model selection was performed over the validation set) and CAP, and better than the Longformer over the entire line. Even though it is based on the mean representation over all words, it appears the relevant information regarding bankruptcy prediction is still sufficiently present. As opposed to the Longformer, the W2V document representations come from in-domain data (i.e., pretrained on 10-k reports).

Finally, we critically evaluate the observed performance improvements for the three-year w.r.t. single-year history setting. The Binary and TF-IDF models are by construction unable to distinguish the different years, but in principle the W2V and Longformer models could learn to capture a deteriorating financial situation over three years of history. However, when evaluating our final W2V models on the test sets with only complete observations (i.e., discard test instances with missing reports), we get the following results. The single year of history AUC is 0.93 (0.94) and the recall@100 is 0.48 (0.36) while the three year history AUC is 0.93 (0.93) and recall@100 was 0.24 (0.28). These results imply that our models taking three years of history into account only lead to better performance metrics as they are able to generate meaningful predictions for companies with some missing reports. Building more expressive models that can leverage the changes in the documents over the years present an interesting avenue for future research.

6 Conclusion and Future Work

Bankruptcy prediction models are valuable in many real-world applications and have received considerable research attention. However, assessing actual progress in the field is not obvious due to the lack of a common benchmark. In this work, we introduce such a benchmark for bankruptcy prediction using textual data along with several baseline models that demonstrate the predictive value of the textual data. We give a detailed discussion on our benchmark and evaluation design choices and share our code to reproduce the experiments.

In future work, we will focus on more advanced models to take into account the temporal evolution of enterprises' financial situation and more advanced language representations (i.e., by finetuning transformer encoders). We also plan to extend the benchmark with structured financial data to build hybrid prediction models.

Acknowledgements

This work was supported by the Research Foundation - Flanders (Grant number G006421N).

References

Edward I Altman. 1968. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The journal of finance*, 23(4):589–609.

Sofie Balcaen and Hubert Ooghe. 2006. 35 years of studies on business failure: an overview of the classic statistical methodologies and their related problems. *The British Accounting Review*, 38(1):63–93.

William H Beaver. 1966. Financial ratios as predictors of failure. *Journal of accounting research*, pages 71–111.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, 24.

Ben S Bernanke. 1981. Bankruptcy, liquidity, and recession. *The American Economic Review*, 71(2):155–159.

Tomasz R Bielecki and Marek Rutkowski. 2013. *Credit risk: modeling, valuation and hedging*. Springer Science & Business Media.

Mark Cecchini, Haldun Aytug, Gary J Koehler, and Praveen Pathak. 2010. Making words work: Using financial text as a predictor of financial events. *Decision Support Systems*, 50(1):164–175.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings.

Tadaaki Hosaka. 2019. Bankruptcy prediction using imaged financial ratios and convolutional neural networks. *Expert Systems with Applications*, 117:287–299.

Kevin Keasey and Robert Watson. 1987. Non-financial symptoms and the prediction of small company failure: A test of argenti's hypotheses. *Journal of Business Finance & Accounting*, 14(3):335–354.

Myoung-Jong Kim and Dae-Ki Kang. 2010. Ensemble with neural networks for bankruptcy prediction. *Expert Systems with Applications*, 37(4):3373–3379.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. *arXiv preprint cs/0205028*.

Lefteris Loukas, Manos Fergadiotis, Ion Androutsopoulos, and Prodromos Malakasiotis. 2021. Edgar-corpus: Billions of tokens make the world go round.

Feng Mai, Shaonan Tian, Chihoon Lee, and Ling Ma. 2019. Deep learning models for bankruptcy prediction using textual disclosures. *European journal of operational research*, 274(2):743–758.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, USA.

Rastin Matin, Casper Hansen, Christian Hansen, and Pia Mølgaard. 2019. Predicting distresses using deep learning of text segments in annual reports. *Expert Systems with Applications*, 132:199–208.

William J Mayew, Mani Sethuraman, and Mohan Venkatachalam. 2015. Md&a disclosure and the firm's ability to continue as a going concern. *The Accounting Review*, 90(4):1621–1651.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space.

Marcus D Odom and Ramesh Sharda. 1990. A neural network model for bankruptcy prediction. In *1990 IJCNN International Joint Conference on neural networks*, pages 163–168. IEEE.

James A Ohlson. 1980. Financial ratios and the probabilistic prediction of bankruptcy. *Journal of accounting research*, pages 109–131.

Tyler Shumway. 2001. Forecasting bankruptcy more accurately: A simple hazard model. *The journal of business*, 74(1):101–124.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

David Veganzones and Eric Séverin. 2018. An investigation of bankruptcy prediction in imbalanced datasets. *Decision Support Systems*, 112:111–124.

Andrey Volkov, Dries F Benoit, and Dirk Van den Poel. 2017. Incorporating sequential information in bankruptcy prediction with predictors based on markov for discrimination. *Decision Support Systems*, 98:59–68.

Gang Wang, Jingling Ma, Gang Chen, and Ying Yang. 2020. Financial distress prediction: Regularized sparse-based random subspace with er aggregation rule incorporating textual disclosures. *Applied Soft Computing*, 90:106152.