Journal of Industrial and Management Optimization

JIMO, 22(8): 3999–4038.
DOI: 10.3934/jimo.2026142
Received: 05 February 2026
Revised: 29 June 2026
Accepted: 10 July 2026
Published: 24 July 2026

https://www.aimspress.com/journal/jimo

Research article

Predicting bankruptcy risk with bidirectional long short-term memory and convolutional neural networks

Ghaneshvar Ramineni, Talayeh Razzaghi* and Kash Barker

School of Industrial and Systems Engineering, University of Oklahoma, Norman, OK 73019, USA

* **Correspondence:** Email: talayeh.razzaghi@ou.edu.

Abstract: Predicting the threat of bankruptcy risk and estimating company survival rate, especially under many challenges that might unfold, is of high economic importance. In this study, we aimed to predict the one-year-ahead and two-year-ahead bankruptcy status of companies using financial features (predictor variables) selected by the recursive feature elimination technique. We proposed the use of deep learning techniques, bidirectional long short-term memory (BiLSTM), and convolutional neural network (CNN) for bankruptcy prediction using temporal data. We compared prediction performance of models using the selected features with the original Altman Z-score financial features. To evaluate the proposed deep learning models, we used a sample of 4,884 companies (150 bankrupt and 4,734 non-bankrupt) drawn from the COMPUSTAT database (1994–2020) and compared them against six benchmark models, reweighted Altman Z-score, linear discriminant analysis, regularized logistic regression, random forest, XGBoost, and LightGBM, over 100 random 70/30 company-level splits, four feature sets, and two prediction horizons. The results showed that no single model dominates across all evaluation criteria: At the conventional decision threshold (t = 0.5), BiLSTM achieved the highest geometric mean of recall and specificity on the Altman feature set (0.653 and 0.615 for the one- and two-year-ahead horizons, respectively), whereas random forest attained the highest AUC on data-driven feature sets (up to 0.790). At that threshold, the tree ensembles flagged almost no failing companies because their calibrated probabilities rarely exceeded 0.5 under severe class imbalance; thus, the proposed deep models, and BiLSTM in particular, were the only ones that detected bankrupt companies reliably without tuning the classification threshold to the data. When the decision threshold was tuned near the bankruptcy base rate, tree-ensemble models recovered comparable balanced

detection, underscoring the importance of evaluating bankruptcy prediction models using multiple performance criteria and operating thresholds rather than a single accuracy metric.

Keywords: bankruptcy prediction; financial distress; deep learning; bidirectional long short-term memory; convolutional neural network; class imbalance; Altman Z-score; feature selection; SHAP analysis
Mathematics Subject Classification: Primary: 68T07; Secondary: 91G40, 62P20

1. Introduction

When a company is in financial distress, owners, employees, and shareholders are adversely impacted, but so are clients that purchase from those companies. An ability to predict the financial distress of a company has wide ranging implications to different stakeholders: From financial institutions making lending decisions, to investors and fund managers making investment decisions, to organizations looking to enter supply contracts. This last implication is especially important to entities with vulnerable supply chain as the financial distress of a company may threaten their performance, and organizations across many industry sectors seek to contract with suppliers that are not at risk of bankruptcy. In 2019, an electric utility, PG&E Corporation in California, filed the biggest bankruptcy of the year losing over $71 billion in assets [1]. With billions of dollars and thousands of jobs at stake, it is crucial to be able to effectively categorize financially healthy and unhealthy institutions.

For the last 100 years, bankruptcy prediction models have played an important role in investment and supplier selection decision-making. At higher levels, these models receive significant attention by policy makers to design proactive policies with the aim of minimizing adverse impacts of bankruptcies on the economy. With today's increasingly complex economic systems and their deep interaction with a network of convoluted stakeholders and competitors, modelers have now access to substantially more relevant features that can help discover hidden financial distress patterns. As a result, more accurate bankruptcy prediction models can be developed to provide early signals of financial distress leading to bankruptcy.

The contributions of this paper are several. First, we identify a large set of financial features using the COMPUSTAT database and identify the critical features relevant to bankruptcy prediction using the recursive feature elimination technique. Our methodological scheme enables us to explore an initial set of 112 candidate features generated from a point in time relative to the occurrence of bankruptcy. Second, we construct two deep learning neural network models, bidirectional long short-term memory (BiLSTM) and convolutional neural network (CNN), which are appropriate for imbalanced time series datasets like those describing supplier health. Last, we demonstrate the effectiveness of the proposed models using COMPUSTAT data and explore the advantage of multiple temporal variations of the features that can discriminate the bankrupt and non-bankrupt companies at different periods prior to bankruptcy. Furthermore, we employ the cost-sensitive learning method embedded with BiLSTM and CNN to treat the imbalance problem associated with few bankrupt companies in the dataset. We

develop bankruptcy prediction models for one year and two years in advance of bankruptcy by applying the proposed techniques to historical financial data collected in advance of the year when bankruptcy was experienced. Most importantly, our multi-criterion evaluation reveals a practical advantage of the proposed deep-learning models that is not captured by AUC alone. At the default decision threshold, the tree-ensemble benchmarks classify almost no companies as bankrupt, whereas the BiLSTM continues to identify failing companies while maintaining a strong balance between recall and specificity. Paired Wilcoxon tests further confirm that this performance difference is statistically significant. These results suggest that the proposed deep-learning models are more dependable in settings where the classification threshold cannot be reliably calibrated for each dataset.

The rest of this paper is arranged as follows: In Section 2, we elaborate on the literature entailing various bankruptcy models over the decades. Section 3 consists of the data analyzed in this study and a brief description of deep learning models. Section 4 includes the implementation of feature selection, the classification results of the models, a discussion of comparative results, and model interpretation using Shapley additive explanations (SHAP) values, followed by concluding remarks and future work in Section 5.

2. Literature review

The initial models that were developed to study the prediction of failure of companies in the early 1920s were based on financial ratio analysis [2]. Ratio analysis helped gain insights into a company's liquidity, efficiency, profitability, and survivability. Factors like the size and industry of the company were considered when comparisons were made among companies [3]. The researchers in [4] compared 13 financial ratios of 19 pairs of successful and failed companies and observed that there existed a consistent change in the ratios at least three years prior to the failure point. The researchers in [5] investigated mean ratios of 183 failed companies for a period of ten years prior to failure and found that the rate of decline of these mean ratios was higher closer to the failure, especially the "current assets to total assets" ratio. The researchers in [6] focused on comparing mean ratios of continuing and discontinued companies between 1926 and 1936 and observed a pattern of increased differences in the ratios as the year of discontinuance approached. However, researchers do not explore the deteriorating trends in the ratios and feature values closer to the bankruptcy point. We intend to capture this essence by exploring one-year and two-year-ahead bankruptcy prediction settings (explained in Section 4.2).

Until the mid-1960s, bankruptcy prediction studies continued to entail ratio analysis and statistical models. Beaver [7] introduced the first univariate model in classifying bankrupt companies, observing that the ratio of net income to total debt had the best predictive ability followed by the ratio of net income to sales. Moreover, Beaver [7] also suggests the use of multiple ratios simultaneously for predicting bankruptcy, leading to the multivariate models. Among the most popular techniques for predicting bankruptcy is the Z-score approach [8], which provides a classification scheme based on five financial characteristics of a company. Such a score, which has seen several iterations since [9,10], provides more of a classification scheme than a means to calculate an implied probability of bankruptcy. A common heuristic suggests that a company with a Z-score < 1.81 is likely to be under financial distress or heading towards bankruptcy. The popular Altman's Z-score, provided in Eq. (1), served as a foundational means to understand the temporal nature of the dataset. The score is a function of five

ratios, found in the first column of Table 1, of seven total variables. The Z-score was developed to describe the bankruptcy risk of companies in the manufacturing sector. The researchers in [11] explored six ratios, in addition to the ones used in calculating Altman's Z-score, that were inspired from [12]. These six additional ratios are found in the second column of Table 1.

$$Z = 1.2A + 1.4B + 3.3C + 0.6D + 1.0E \tag{1}$$

For companies outside the manufacturing sector, an update to the Z-score [10], denoted Z'', is provided in Eq. (2). A Z''-score < 1.1 implies that a company is under financial distress. The first four variables from Table 1 are used in the calculation of the Z''-score.

$$Z'' = 6.56A + 3.26B + 6.72C + 1.05D \tag{2}$$

Table 1. Eleven variables analyzed by [11], including the five ratios used in the calculation of Altman's Z-score.

Variables from [8]	Variables from [12]
$A = \frac{\text{net working capital}}{\text{total assets}}$	$F = \frac{\text{earnings before interest and taxes}}{\text{sales}}$
$B = \frac{\text{retained earnings}}{\text{total assets}}$	$G = \frac{\text{total assets}_t - t\text{otal assets}_{t-1}}{\text{total assets}_{t-1}}$
$C = \frac{\text{earnings before interest and taxes}}{\text{total assets}}$	$H = \frac{\text{sales}_t - \text{sales}_{t-1}}{\text{sales}_{t-1}}$
$D = \frac{\text{market value of shares} \times \text{number of shares}}{\text{total debt}}$	$I = \frac{\#\text{ employees}_t - \#\text{ employees}_{t-1}}{\#\text{ employees}_{t-1}}$
$E = \frac{\text{sales}}{\text{total assets}}$	$J = \left(\frac{\text{net income}}{\text{stockholder equity}}\right)_t - \left(\frac{\text{net income}}{\text{stockholder equity}}\right)_{t-1}$
	$K = \left(\frac{\text{market value/share}}{\text{book value/share}}\right)_t - \left(\frac{\text{market value/share}}{\text{book value/share}}\right)_{t-1}$

With increasing efficiency of data collection, the availability of machine learning models has provided newfound sophistication and accuracy in predicting bankruptcy, at the cost of creating complex and less explainable models. While machine learning models help improve the accuracy of the predictions, feature selection techniques are especially useful in narrowing the vast number of variables that might be involved. Techniques that have been applied to bankruptcy prediction for several financial variables include neural networks [13–17], regression techniques [18], stochastic processes [19], over sampling with neural networks [20,21], feature selection engineering [22], and regression trees [23–25], among others [11,26–31]. Some studies have inferred that a set of limited features is sufficient in predicting bankruptcy accurately [11,32]. The researchers in [11] developed several bankruptcy prediction models for data obtained from 10 types of industries and achieved an AUC of 92.97% for boosting, 92.48% for bagging, and 92.92% for random forest models, using just

11 attributes/features. However, very few of these researchers [23,15] address the influence of dynamic features using the time series data in their prediction models. The researchers in [33] and [34] reviewed the progress of a wide range of bankruptcy models in the past few decades.

Researchers have expanded the methodological approaches for bankruptcy prediction in several directions. The researchers in [35] introduced the Omega Score, which is particularly well-suited for small and medium-sized enterprises, and demonstrated that combining "hard" financial information with "soft" qualitative information can improve default prediction performance. The researchers in [36] provided a foundational discussion of the distinction between hard and soft information in financial decision-making. In parallel, deep learning methods have continued to evolve, including ensemble-based and neural network architectures for bankruptcy and default prediction across company-size segments [37,38]. Researchers have also documented the direct and indirect costs associated with bankruptcy [39,40], while others have shown that these effects may propagate through suppliers and trading partners, amplifying broader economic consequences [35].

Motivated by [11], integrating dynamic predictor variables into prediction models can improve prediction accuracy. In addition, bankruptcy may be related to many explanatory variables (features) that other studies have shown to be effectively modeled by non-linear techniques [11,41]. Advanced machine learning techniques, such as neural networks and deep learning, have been successful in not only modeling non-linear decision boundaries but also modeling data with sequential predictor variables (such as time series). Two popular variants of deep learning methods that are especially effective for time series data are long short-term memory (LSTM) neural networks and CNN. LSTM is a recurrent neural network model built upon feedback connections, and has been very successful for time series prediction and classification problems [20,42–44]. In this study, we employ a bidirectional LSTM model (BiLSTM) that processes the input in two directions from past time periods to future periods and vice versa. CNN represents a class of deep neural networks with multiple layers between the input and output layers, employing the mathematical function of convolution in its operation. It has been used for many applications among facial recognition [45], image classification [46], fraud detection [47], speech recognition [48], and time series analysis [49]. Moreover, the researchers in [50] used thte SHAP technique to interpret their bankruptcy prediction results.

Studies have been conducted on financial analysis and bankruptcy prediction using neural networks and deep learning methods [17,51–54]. Moreover, the researchers in [41] focused on using deep learning models for bankruptcy prediction using textual disclosures. The researchers in [55] provided a review on the different machine learning and neural network models used in bankruptcy prediction. Limitations of these works include (i) not considering the impact of dynamic features to represent temporal company performance and (ii) the treatment of imbalance issues in the dataset. Since the number of companies that undergo bankruptcy is very small relative to those that do not, it is expected that a bankruptcy dataset would be imbalanced. As such, it is imperative that this imbalance is accounted for when building a predictive model. In addition, there have not been adequate computational experiments that evaluate the applicability of deep learning algorithms to handle imbalanced data in bankruptcy prediction problems. Building on this literature, we systematically compare BiLSTM and CNN models against a reweighted Altman Z-score framework and several modern machine learning benchmarks using a common company-level dataset.

3. Data and methods

In this section, we provide background on the data used for bankruptcy prediction, as well as the two deep learning neural network methods along with the feature selection technique used in the analysis.

3.1. Data

The data used in this analysis are obtained from Standard & Poor's (S&P) COMPUSTAT database. The COMPUSTAT database contains fundamental financial, statistical, and market information for active and inactive publicly held companies from around the world. For most companies, the database has annual data since 1950 and quarterly data since 1962. The annual data include all financial data from a company's Form 10-K, including company descriptions, balance sheet items, income statement data, and cash flow information, as well as standard industry classifiers, market identifiers, and S&P proprietary data. Some examples of these variables include a business description, stock ticker, market value of shares, cash on hand, and net sales. Similarly, the quarterly data include all data from Form 10-Q submissions, like total assets, total liabilities, and retained earnings. In total, there are 974 non-screening variables for annual data and 674 for quarterly data. The COMPUSTAT database is updated daily. For this analysis, annual data from the North American companies, mostly from the United States of America and Canada, ranging from 1994 to 2020, are used. The data are queried based on a list of Standard Industrial Classification (SIC) codes, which the Security Exchange Commission uses to classify company industries. The sectors considered for the analysis are shown in Table 2.

Table 2. List of the industry sectors and their corresponding SIC codes.

Range of SIC codes	Industry
A: 0100–0999	Agriculture, Forestry, and Fishing
B: 1000–1499	Mining
C: 1500–1799	Construction
D: 2000–3999	Manufacturing
E: 4000–4999	Transportation, Communications, Electric, Gas, and Sanitary Service
F: 5000–5199	Wholesale Trade
G: 5200–5999	Retail Trade

The bankrupt label is assigned to companies that have filed for bankruptcy (e.g., bankruptcy Chapter 7 occurs when a company's assets are liquidated and the company ceases to exist, or Chapter 11, or "reorganization bankruptcy," occurs when a company is reorganized to a different entity while continuing to operate).

A company is identified as bankrupt using the COMPUSTAT deletion reason (*dlrsn* codes 2 or 3) or a status alert (*stalt* equal to "TL"). Because these flags do not always coincide with the exact year that a company fails, we further classify flagged companies by the pattern of the flag across the company's records to assign a reliable bankruptcy year. When the flag appears only on the company's

final record, the flag is treated as marking the actual bankruptcy event and that year is taken as the bankruptcy year. When the flag is present on every record of a company that subsequently disappears from the database, the company's last available year is taken as the bankruptcy year. Companies whose flag appears in the middle of their history but that continue reporting afterward are excluded from the bankrupt set, as their status cannot be unambiguously associated with a single failure event; retaining them would risk contaminating the labels. This procedure yields a clean set of bankrupt companies with well-defined bankruptcy years, which is essential for constructing the pre-event input windows described in Section 4.2.

Figure 1 represents the average Z-score plot for 7572 manufacturing companies (295 bankrupt and 7277 non-bankrupt) on the vertical axis and a time reference on the horizontal axis. The time reference depicts the most recent year of company data on the right-hand side. For non-bankrupt companies, the right-hand side represents the most recent available observation from the database, while for bankrupt companies, the right-hand side represents the last observation before the bankruptcy event. Time prior to the most recent year extends to the left. For example, for a bankrupt company, three time periods from the right represents three years prior to bankruptcy. Bankrupt companies are represented with red, and non-bankrupt are represented with green. The data for bankrupt companies after the bankrupt point are ignored. This is done to maintain uniformity among the bankrupt companies as few companies reorganize their financials and continue to operate, sometimes as a different entity, while others cease to exist altogether. From Figure 1, one can observe that there is a distinct separation of average Z-score between the bankrupt and non-bankrupt companies in the manufacturing sector, especially close to the bankruptcy point. This separation motivates the use of time series methods to predict bankruptcy.

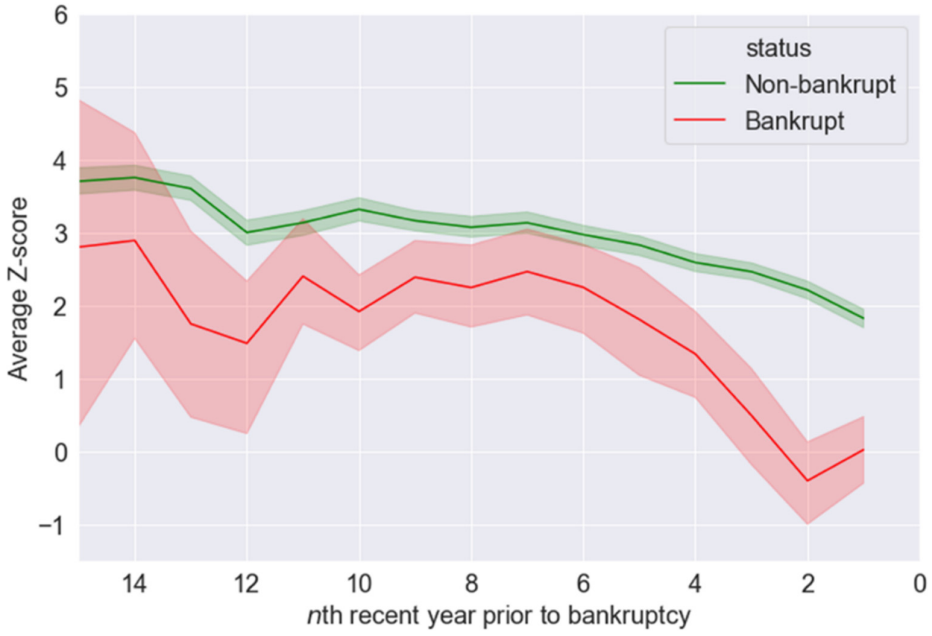

Figure 1. Average Z-score plot for manufacturing companies with Z-score in range [−15,15].

Table 3 provides an overview of the data, consisting of information about companies from 1994–2020. The unit of measurement of some features such as 'lo', 'lt', 'ebit', and 'dltt' is $ million and that of 'growth in emp' and 'sale' is in units. The inf and -inf values occur when the denominator is a 0 while computing its value. We can observe that some features such as 'adjex_f', 'ebit', and 'cogs' are highly skewed. This is because the data includes companies ranging from small to large.

Table 3. A statistical review of the data for a set of selected features.

Feature	Count	Mean	Std	Min	25%	50%	75%	Max
Change in return on equity	161732	-	-	−∞	0.077	0.793	1.284	inf
Other liabilities, total (lo)	180589	336.419	2186.952	−27.06	0	1.641	40.9	104481
Cumulative adjustment factor, fiscal (adjex_f)	172759	1.277	4.28	0	1	1	1	822.5
Total liabilities (lt)	180434	2118.845	10378.5	0	7.355	60.614	606.192	460442
Market value of equity / Total liabilities (D)	107394	inf	-	0	2.236	9.813	inf	inf
Income before extraordinary items (ib)	180019	128.114	1119.308	−85162.2	−4.702	0.888	31.371	104821
Earnings before interest and taxes (ebit)	180357	833.515	4001.36	−0.023	0	8.621	250.254	207174
Long-term debt, total (dltt)	132859	8.863	42.563	−1	−0.878	−0.125	3.766	2299
Growth in emp	179061	257.174	1426.221	−25913	−2.141	4.249	72.381	71230
Leverage	155837	inf	-	0	0.819	2.111	6.369	inf
Net sales / turnover (sale)	179799	2508.935	12858.86	−1965	9.8685	121.15	883.9	521426
Cost of goods sold (cogs)	179748	1745.547	9894.701	−366.645	7.643	75.325	574.048	435726.3

Among the constructed variables, leverage is defined as the ratio of total liabilities to total assets (lt/at), a book-value measure of financial leverage. This definition is adopted in place of a market-based leverage measure so that the variable can be computed consistently for all companies from balance-sheet items alone, without requiring market-price data that are unavailable or unreliable for a portion of the sample. Using the book-value definition also keeps the leverage measure consistent with the other accounting-based ratios used in the study.

Missing values are common in the COMPUSTAT database because not all companies report every item in every year. To preserve each company's temporal structure while avoiding the introduction of information from other companies, missing values are imputed using a five-stage cascade applied in order of increasing scope. First, a missing value is filled using the company's own mean for that variable across its available years. If the variable is still missing for a company for all years, it is imputed using the mean of companies sharing the same four-digit Standard Industrial Classification (SIC) code, provided at least five companies are available in that group. The cascade

then falls back, in turn, to the three-digit SIC mean and the two-digit SIC mean for any values that remain missing. Finally, any values still missing after the industry-level stages are filled with the global median of the variable. This ordered cascade prioritizes company-specific information and falls back to progressively coarser industry groupings only, when necessary, thereby limiting the extent to which imputation blurs company-level financial trajectories.

3.2. Feature selection

3.2.1. Recursive feature elimination with cross-validation

The recursive feature elimination with cross-validation (RFECV) method is used for feature selection in this study. RFECV extends the traditional recursive feature elimination (RFE) approach proposed in [56] by automatically determining the optimal number of features through cross-validation rather than requiring the feature count to be predefined. The method recursively ranks features according to their importance and removes the least important features at each iteration. In this study, a random forest classifier is used as the ranking estimator. At every elimination step, model performance is evaluated using 5-fold cross-validation with ROC-AUC as the scoring metric, which is more appropriate for imbalanced bankruptcy prediction tasks. The feature subset corresponding to the highest cross-validated ROC-AUC score is selected as the optimal feature set. To avoid information leakage, RFECV is performed independently within each training split during the experimental procedure.

3.2.2. Cumulative-importance method

Because the number of features retained by RFECV can vary considerably from one training split to another, we employ a second, complementary selection method based on a cumulative feature-importance threshold (hereafter, the cumulative-importance method). For each training split, a random forest is fitted to the time-averaged features, and the resulting Gini importances are ranked in descending order. Features are then retained, in order of importance, until their cumulative importance reaches a fixed threshold of 0.75, so that the smallest set of features accounting for 75% of the total importance is selected. Unlike the cross-validation peak used by RFECV, the cumulative-importance criterion does not depend on locating the maximum of a relatively flat cross-validation curve and therefore tends to yield a more stable number of selected features across splits. Reporting both methods enables us to assess the robustness of the selected features and to distinguish features that are consistently informative from those whose selection is sensitive to the particular method or training sample.

3.3. Long short-term memory networks

Long short-term memory (LSTM) is an extended variation of the recurrent neural network (RNN) architecture, and was proposed in [57]. In this section, the model of LSTM architecture for bankruptcy prediction is introduced. Unlike RNN, LSTM is powerful in handling the vanishing gradients and learn

long-term dependencies in modeling the time series using memory cells [58,59]. A memory cell includes four units: an input gate, an output gate, a forget gate, and a self-recurrent neuron. The interactions between neighboring memory cells and the memory cells are controlled by the gates. The input gate controls how much of the incoming input signal is used to update the memory cell state. The output gate determines how much of the cell state is passed to the hidden state and subsequently influences other memory cells or network components. Furthermore, the previous state of the memory cell will be remembered or forgotten by the forget gate. The mathematical formulation of LSTM in Figure 2 is adapted from [60]. Assume that a dataset is represented by a set $D = \{(X_l, y_l)\}$, where $l = 1, 2, \dots, n$, and $X_l = (x_1, x_2, \dots, x_T)_l$ represent an input sequence for sample l for an LSTM, $x_t \in \mathcal{R}^m$ is the m-dimensional input vector to the memory cell at time t, and $y_l \in \{0, 1\}$ is the associated class label; W_i, W_f, W_c, W_o, U_i, U_f, U_c, U_o, and V_o are weight matrices; b_i, b_f, b_c, and b_o are bias vectors; and h_t is the value of the memory cell at time t. We calculate the forward pass of the neural network through the following equations that describe LSTM cell vectors at time t. The values of the input gate (i_t) and the candidate state ($\tilde{C}_t$) of the memory cell at time t can be calculated with Eqs. (3) and (4).

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{4}$$

The forget gate (f_t) and the state of the memory cell (C_t) at time t are formulated with Eqs. (5) and (6).

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{5}$$

$$C_t = i_t \odot \tilde{C}_t + f_t \odot C_{t-1} \tag{6}$$

Finally, the output gate (O_t) and memory cell (h_t) at time t are calculated with Eqs. (7) and (8).

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o C_t + b_o) \tag{7}$$

$$h_t = o_t \odot \tanh(C_t) \tag{8}$$

In the above, $\odot$ represents element-wise multiplication, and σ and tanh are non-linear logistic sigmoid and hyperbolic tangent activation functions, respectively.

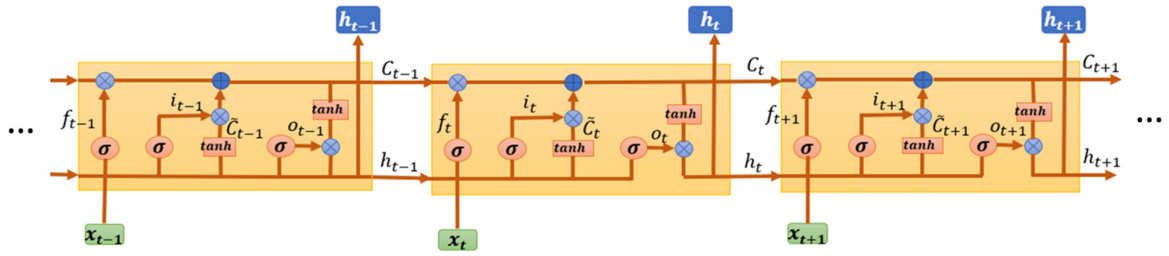

Figure 2. The sequential LSTM architecture overview. $\otimes$ and $\oplus$ denote element-wise multiplication and summation, respectively (adapted from [60]).

3.4. Bidirectional LSTM

The classic LSTM is a unidirectional network that can process data only following one direction using past information. However, past and future company status and attributes (temporal sequences) provide valuable information for accurate prediction of bankruptcy. Bidirectional LSTM (BiLSTM) is a well-known variant of LSTM first introduced in [61] that memorizes past and future information. BiLSTM consists of two hidden layers that are connected to the input and output. The sequence of operations in the first (forward) layer conforms to the same direction of the data sequence, while the other (backward) layer operates in the reverse direction. This structure enables the model to use additional information to process earlier time steps in reverse. Furthermore, BiLSTM has been successful in handling sequence data in many areas, including time series classification [60], natural language processing [62], speech recognition [63], and handwriting recognition [64]. Moreover, studies have shown that BiLSTM performs better than LSTM for some applications that include time series data [44]. Therefore, BiLSTM is used in this study.

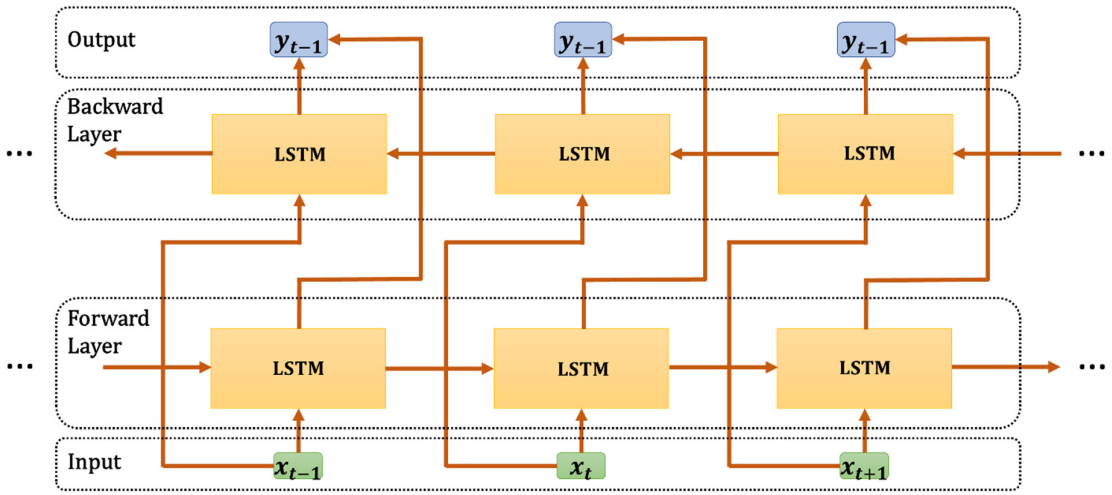

Figure 3. The sequential BiLSTM architecture overview adapted from [65].

Figure 3 shows the methodology followed to construct the BiLSTM network. After comparing the performance of multiple architectures, the best performance is achieved by the BiLSTM architecture shown in Figure 4.

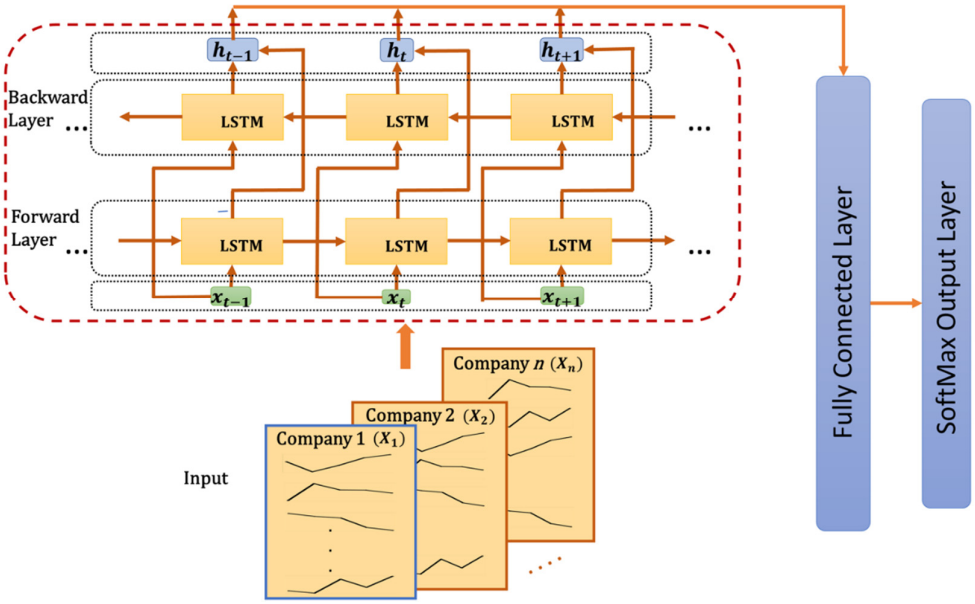

Figure 4. The architecture of the BiLSTM model for bankruptcy prediction.

3.5. Convolutional neural networks

A CNN is a modification of the conventional artificial neural network (ANN), which has been successful in extracting high-level features from image [66], text [67], and time series [68] data. In general, the steps involved in building a CNN model include (i) a convolution operation followed by activation function, (ii) a downsampling operation using pooling layers, (iii) flattening the pooled features, and (iv) computing a final score function using full-connected layers (i.e., conventional ANN layers). After feeding the input data into a convolutional layer, the local feature representation of data is created. Each convolutional layer is made up of multiple same collections of the identical neurons, which are called filters (or kernels). Due to local connectivity of each neuron with only a subset of inputs, the CNN results in a model with sparse interactions and thus a reduced number of parameter sharing [67]. Like ANN, an activation function is applied to each neuron in the convolutional layer to model the non-linearity in the output. For the activation function, we utilize the rectified linear unit (ReLU) function for all convolutional layers, which is defined in Eq. (9).

$$g(x) = \max(0, x) \tag{9}$$

The most significant benefit of ReLU is that it is much faster in training compared to other activation functions [69]. As our objective is to predict the bankruptcy status of companies using multivariate time series data, we develop a CNN model for this problem. Our CNN model is inspired by the notation introduced in [70]. We illustrate the architecture of our proposed CNN in Figure 5**,** which is identified through extensive experiments for the appropriate number of filters, the size of the filters, and the number of pooling layers.

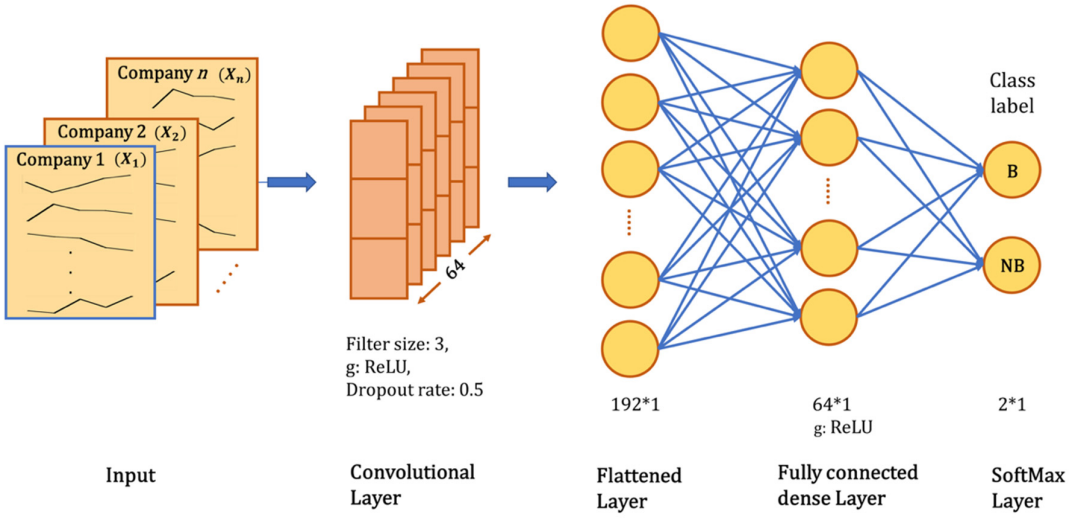

Figure 5. The architecture of our CNN model for bankruptcy prediction.

3.6. Performance measures

Researchers have used several measures to evaluate how well the model predicts the bankruptcy status of each company [11,41,71]. These measures are computed from the confusion matrix in Table 4, where TP refers to the number of *true positive* company classifications, FP refers to *false positive* classifications, FN refers to *false negative* classifications, and TN refers to *true negative* classifications. TP and TN represent the number of accurately predicted bankrupt and non-bankrupt companies, respectively. FN represents the number of companies predicted as non-bankrupt, while the actual status of the companies is bankrupt. On the contrary, FP represents the number companies predicted as bankrupt when their actual status is non-bankrupt. It is important to focus on the FPs and FNs in such analyses, as they can impact the companies' future. For example, if a company is labeled FP, it can have a negative impact on the business of the healthy company. That is, if a company is predicted to be at risk for bankruptcy despite being a financially healthy one and such a prediction is made public, it could impact the decisions of the stakeholders or influence the investors' faith in the company. Similarly, if a company is wrongly predicted as FN, it can lead to substantial loss of funds and affect stakeholders. The measures include the area under the receiver operating characteristics curve (AUC) together with sensitivity (or recall), specificity, G-mean, F-measure, precision, and accuracy, the calculation of which are found in Eqs. (10)–(15), respectively. Sensitivity represents the ratio of accurately predicted positive samples to all positive samples, while specificity is the ratio of accurately predicted negative samples to all negative samples. Precision is the ratio of accurately predicted positive samples to the total predicted positive samples. Sensitivity and precision focus on the bankrupt companies, while specificity focuses on the non-bankrupt companies. Accuracy represents the ratio of accurately predicted status (positive and negative samples) to the total number of companies, giving us insight into the overall prediction performance.

Table 4. Confusion matrix for binary classification.

Predicted	Actual	
	Bankrupt	Non-bankrupt
Bankrupt	TP	FP
Non-bankrupt	FN	TN

$$\text{Sensitivity (or Recall)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{10}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{11}$$

$$\text{G} - \text{mean} = \sqrt{\text{sensitivity} \times \text{specificity}} \tag{12}$$

$$\text{F1} - \text{score} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{13}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{14}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{15}$$

In addition to the metrics defined in Eqs. (10)–(15), we report four additional performance measures that complement the original evaluation set. Type I error (false positive rate) and type II error (false negative rate) separately quantify the two distinct, cost-relevant errors in bankruptcy prediction rather than combining them into a single measure. We also report the Matthews correlation coefficient (MCC), a balanced correlation-based metric derived from the confusion matrix that is robust to class imbalance and ranges from −1 to +1, where 0 indicates chance-level performance. The Brier score evaluates the calibration accuracy of predicted probabilities by measuring the mean squared error between the predicted probabilities (p_i for sample i) and the observed binary outcome (y_i for sample i). Unlike AUC, which evaluates only ranking performance, the Brier score also captures the quality of probability calibration. These four additional measures are formally defined in Eqs. (16)–(19).

$$\text{Type I error} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{16}$$

$$\text{Type II error} = \frac{\text{FN}}{\text{FN} + \text{TP}} \tag{17}$$

$$\text{MCC} = \frac{(\text{TP} \cdot \text{TN}) - (\text{FP} \cdot \text{FN})}{\sqrt{(\text{TP}+\text{F})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}} \tag{18}$$

$$\text{Brier} = \frac{1}{\text{n}} \sum_{\text{i}=1}^{\text{n}} (\text{y}_\text{i} - \text{p}_\text{i})^2 \tag{19}$$

3.7. Benchmark models

To provide a rigorous comparison, we evaluate the proposed BiLSTM and CNN against six benchmark models trained on the same company-level train-test splits and feature sets. The benchmark models include: (i) A reweighted Altman Z-score, implemented using logistic regression with the original five Altman ratios, addresses the concerns that the original 1968 weighting scheme may not adequately reflect contemporary economic conditions; (ii) linear discriminant analysis (LDA), which represents the classical statistical approach underlying the original Altman Z-score model; (iii) regularized logistic regression with L1 or L2 penalties; (iv) random forest; (v) XGBoost; and (vi) LightGBM. For the proposed deep learning models, the data are represented in their natural three-dimensional structure (company × timesteps × features). In contrast, the benchmark models receive the same underlying information with the temporal dimension flattened into a single feature vector of length (timesteps × features) for each company. To ensure consistency across all experiments, all models are trained using class-weighted learning to address class imbalance, matching the cost-sensitive strategy adopted for the BiLSTM and CNN models.

- Regularized logistic regression is a generalized linear classifier that models the log-odds of bankruptcy as a linear combination of the input features. We use L1 (lasso) and L2 (ridge) penalties to shrink the coefficient estimates and reduce overfitting, with the regularization strength selected using the validation set. The reweighted Altman Z-score model is implemented as logistic regression restricted to the original five Altman ratios; this enables the relative importance of the ratios to be re-estimated on the data rather than being fixed at the 1968 weights.
- Linear discriminant analysis (LDA) is the classical statistical method underlying the original Altman Z-score model. LDA assumes that the feature vectors of each class are drawn from multivariate normal distributions with a common covariance matrix and assigns a company to the class with the higher posterior probability. The decision boundary is therefore linear in the features, and the model can be interpreted as a parametric counterpart to logistic regression under the normality assumption.
- Random forest is an ensemble of decision trees in which each tree is grown on a bootstrap sample of the training data, and each split considers only a random subset of the features. The final prediction is obtained by averaging the predicted class probabilities across trees. Random forests are robust to feature scaling, handle non-linear interactions naturally, and provide an internal measure of feature importance based on the impurity reduction at each split, which is used here as the basis of the cumulative-importance feature-selection method.
- XGBoost [72] is a gradient-boosted tree ensemble that builds trees sequentially, with each tree fitted to the residual errors of the previous ensemble. It uses a second-order Taylor expansion of the loss together with regularization on the tree structure and supports a *scale_pos_weight* parameter that we set to the ratio of negative to positive samples to address the class imbalance.
- LightGBM [73] is a fast gradient-boosting implementation that grows trees leaf-wise rather than level-wise and uses histogram-based feature binning. Like XGBoost, it supports class weighting via an *is_unbalance* or *scale_pos_weight* setting; we use class weights proportional to the inverse class frequencies, matching the cost-sensitive strategy applied to the other models.

3.8. Hyperparameter tuning

For each machine learning model (BiLSTM, CNN, logistic regression, random forest, XGBoost, and LightGBM), hyperparameters are selected using the grid search. Specifically, the dataset is first divided using a company-level 70/30 train-test split to avoid information leakage across companies. Within the training portion, 15% of the data is further reserved as a validation set for hyperparameter tuning. The hyperparameter configuration that maximizes the validation AUC is selected and subsequently fixed across all 100 evaluation seeds used in the main experiments. Hyperparameter tuning is performed separately for each model and feature-set-size configuration. Linear discriminant analysis has no tunable hyperparameters and uses default settings. To reduce overfitting in the deep learning models, early stopping is employed using validation AUC as the monitoring criterion with a patience of 20 epochs. This procedure also serves as an implicit regularization mechanism during training.

The hyperparameter value ranges searched for each model are summarized in Table 5. For every model and feature-set configuration, all combinations in the corresponding grid are evaluated, and the configuration with the highest validation AUC is obtained.

Table 5. Hyperparameter range for each model.

Model	Hyperparameter	Values searched
BiLSTM	Units; dropout; batch size; learning rate	{16, 32, 64}; {0.2, 0.3, 0.5}; {16, 32}; {0.001, 0.0005}
CNN	Filters; kernel size; dropout; batch size; learning rate	{32, 64, 128}; {2, 3}; {0.2, 0.3, 0.5}; {16, 32}; {0.001, 0.0005}
Logistic regression	Inverse regularization C; penalty	{0.01, 0.1, 1.0, 10.0}; {L1, L2}
Random forest	Number of trees; maximum depth; minimum samples per leaf	{100, 200, 500}; {none, 5, 10, 20}; {1, 2, 5}
XGBoost	Number of trees; maximum depth; learning rate	{100, 200, 500}; {3, 4, 6}; {0.05, 0.1, 0.2}
LightGBM	Number of trees; number of leaves; learning rate	{100, 200, 500}; {15, 31, 63}; {0.05, 0.1, 0.2}
LDA	None (default settings)	—

4. Results and discussion

4.1. Feature selection

To develop accurate prediction models, the first step is to perform feature selection. In Section 3.1, we discussed two sets of features: The five ratios that made up the [8] Z-score and the expanded set of 11 ratios (inclusive of the five from Altman) used by [11], all of which are in Table 1. We note that the features from [8] and [11] are developed from a financial perspective.

RFECV is performed independently within each training split of the experimental procedure. For benchmarking purposes, we evaluate all models using two fixed feature sets from the literature: The original five financial ratios proposed by [8] and the eleven features used by [11]. Feature selection is

conducted on time-averaged variables to ensure comparability between the deep learning models, which preserve the temporal structure of the data, and the benchmark models, which use flattened time-series representations.

In this work, we focus our analysis and comparative evaluations based on four sets of features: (i) Five features from [8], (ii) 11 features from [11], (iii) the RFECV-selected features, and (iv) the cumulative-importance-selected features, both drawn from an initial pool of 112 candidate features by the two data-driven selection methods described in Section 3.2.

Table 6. Most frequently selected features by maximum selection frequency across the two feature-selection methods on the one-year-ahead horizon. Frequency is the percentage of 100 random 70/30 splits in which the feature is selected; average rank is the mean impurity-importance rank was selected.

Feature	Cumimp freq (%)	Cumimp avg rank	RFECV freq (%)	RFECV avg rank
Working capital / Total assets (Altman A)	100	34.49	54	26.09
Retained earnings / Total assets (Altman B)	100	35.49	76	23.39
EBIT / Total assets (Altman C / productivity)	100	36.49	96	21.07
Market value of equity / Total liabilities (Altman D)	100	37.49	61	28.20
Common shares outstanding	100	10.63	77	5.47
Common shares used in basic EPS	100	11.63	85	5.99
Price close, fiscal year-end	100	33.49	84	20.32
Net income / Common equity	100	50.82	70	35.61
Growth of total assets	100	48.07	63	34.37
Income before extraordinary items	100	22.60	100	12.07
Earnings before interest and taxes (EBIT)	100	17.26	88	9.45
Earnings per share, basic incl. extraordinary	100	19.04	100	9.82
EBIT / Net sales (operating margin)	100	47.07	99	27.48
Retained earnings	100	27.79	71	17.94

Because feature selection is now performed independently within the training portion of each of the 100 random splits, the selected subsets vary from one split to another. To characterize this variability, we record, for each candidate feature, the proportion of the 100 splits in which it is selected (its selection frequency) together with its average importance rank when selected. The two selection methods exhibit markedly different stability profiles. The cumulative-importance method retains a broad and consistent set of variables: A total of 35 features are selected in at least 90% of the splits, of which 16 are selected in every split, including the Altman ratios and core earnings, profitability, and scale measures (e.g., EBIT, retained earnings, operating margin, and productivity). In contrast, RFECV is far more selective and less stable, retaining only five features in at least 90% of the splits. Importantly, both methods converge on a common high-frequency core, namely the Altman ratio of EBIT to total assets together with income before extraordinary items, earnings per share, operating margin, and productivity, each selected in at least 96% of the splits under both methods. The agreement of two procedurally distinct selection methods on this small set of earnings- and profitability-related variables indicates that these features are robustly informative for bankruptcy prediction rather than artifacts of

a particular selection rule or training sample. Table 6 presents the features with the highest selection frequencies across the two feature-selection methods for the one-year-ahead prediction horizon.

4.2. BiLSTM and CNN models

Figure 6 shows the overview of our method. The output variable is defined as a binary company-level label, where bankrupt companies are assigned a value of 1, and non-bankrupt companies are assigned a value of 0. Consequently, the model produces one predicted bankruptcy probability per company using its corresponding historical input window. Within this single dataset, we evaluate two prediction horizons: (i) One-year-ahead, using input years $t-4$ through $t-1$, and (ii) two-year-ahead, using input years $t-4$ through $t-2$. The bankruptcy year (year t) is never used as input.

Cleaning the data includes eliminating features with more than 10% missing values, and highly correlated features (with an absolute value of the Pearson correlation coefficient, r, greater than 0.95, chosen to avoid highly redundant information). After this, missing values are imputed using the five-stage company- and industry-level cascade described in Section 3.1. The samples are drawn from companies that either filed or did not file for bankruptcy during the period 1994–2020. After applying the 5-year consecutive-data filter and the cleaning steps described above, the final dataset consists of 4,884 companies, including 150 bankrupt and 4,734 non-bankrupt companies, as reported in Table 7. This corresponds to a bankruptcy imbalanced ratio of approximately 3.1% (b = 150/4,884 ≈ 0.031) and a non-bankrupt-to-bankrupt ratio of about 31.6 to 1. This high degree of class imbalance is typical in bankruptcy prediction and motivates the use of cost-sensitive training, described below, as well as imbalance-aware evaluation metrics, discussed in Section 4.3.

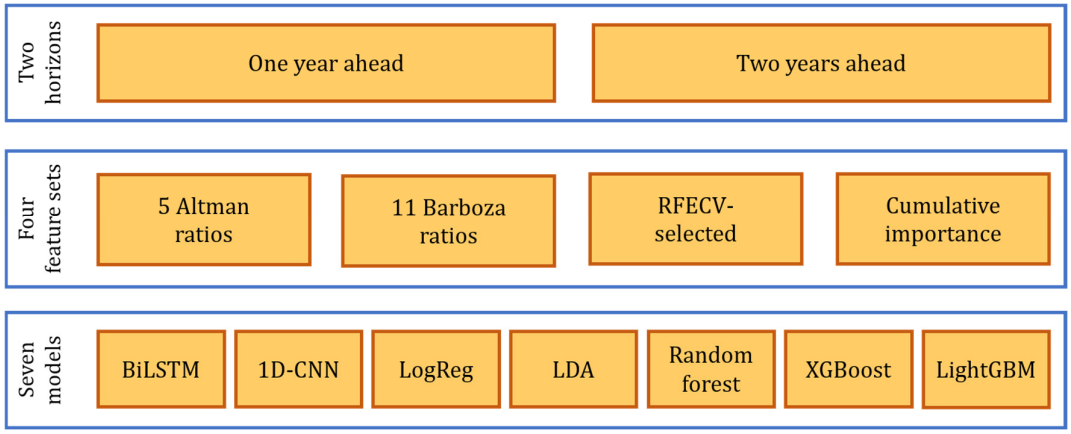

Figure 6. Overview of the experimental design: 4,884 companies split 70/30 (company-level) over 100 random seeds, evaluated across 4 feature sets, 7 predictive models, and 2 prediction horizons. Class imbalance handled via cost-sensitive learning.

The data are then partitioned into training and test sets using a 70/30 company-level split. The split is performed at the company level so that all annual records for a given company are assigned together to either the training or test set, preventing records from the same company from appearing

in both sets. Under this split, the training set contains 3,418 companies, including 105 bankrupt and 3,313 non-bankrupt companies, while the test set contains 1,466 companies, including 45 bankrupt and 1,421 non-bankrupt companies. The same split sizes apply to the one-year-ahead and two-year-ahead prediction settings, and Table 7 summarizes the corresponding training and test sample sizes.

Table 7. Number of companies in the train and test sets of the dataset.

	Bankrupt	Non-bankrupt	Total
Training (70%)	105	3,313	3,418
Testing (30%)	45	1,421	1,466
Total	150	4,734	4,884

Class imbalance is addressed during training through cost-sensitive learning. For the linear and tree-based models, the *'class_weight'* parameter is set to *'balanced'* following the Scikit-learn convention, where the weights for the positive (bankrupt) and negative (non-bankrupt) classes are computed as $n/(2n^+)$ and $n/(2n^-)$, respectively, with n^+ and n^- denoting the numbers of positive and negative samples. For XGBoost, '*scale_pos_weight'* is set to the ratio of negative to positive samples, while explicit class weights are provided for the BiLSTM and CNN models. The prediction on the test data provides the (complementary) probabilities of companies being bankrupt or non-bankrupt. These probabilities are then converted to a binary representation of 1 for bankrupt companies and 0 for non-bankrupt companies using a threshold probability of 0.5.

4.3. Model evaluation and discussion

In this section, we explain the experimental results using BiLSTM and CNN in combination with different sets of features per different temporal dataset (as stated in Section 4.2). We implement our methods using Python version 3.7.4 with Keras [74] and TensorFlow libraries [75] on an Intel core i7, 3.4 GHz with 16 Gb of RAM in a 64-bit platform.

For comparability with other studies, we analyze the original raw data without applying transformation (e.g., normalization, following [11,13,14,16]). This is done to retain the originality of the data samples and to test the predictive capability of the models without depending on transformation techniques. As we focus on a wide range of companies, this approach also helps in preserving the divergence in magnitudes of the attributes' values. Table 8 and Table 9 report the average values of the performance measures over 100 random 70/30 splits per feature set and prediction horizon. For the sake of consistency and robustness of the results, all performance metrics, including precision, recall, specificity, AUC, F-measure, and accuracy, for BiLSTM and CNN models are an average over 100 iterations of the same data with different random seeds (bootstrap), as shown in Table 8 and Table 9. We observe from these results that the ranking of the models depends highly on the evaluation criterion. When models are ranked by AUC, the tree ensembles perform best. At the one-year-ahead horizon, random forest achieves the highest AUC on the data-driven feature sets, with scores of 0.790 for the cumulative-importance set, followed by 0.781 for the RFECV set, as shown in Table 8. XGBoost and LightGBM achieve comparable results, whereas BiLSTM reaches an AUC of 0.713 on the Altman feature set. However, AUC measures only the quality of a model's probability

ranking and does not reflect the decisions a model makes at a given probability threshold. At the conventional decision threshold of 0.5, the pattern is reversed: The tree ensembles classify almost no companies as bankrupt. For example, random forest attains a recall near zero and a type II error close to one across all feature sets and both prediction horizons because their predicted probabilities rarely exceed 0.5 under such severe class imbalance. In contrast, BiLSTM remains an effective classifier at this threshold. Using the Altman feature set, it achieves the highest geometric mean of recall and specificity among all models, with values of 0.653 and 0.615 for the one- and two-year-ahead horizons, respectively. It also maintains relatively high recall, 0.629 and 0.600, with correspondingly lower type II errors of 0.371 and 0.400 (See Table 8 and Table 9). Among temporal datasets, the two-year-ahead dataset shows the lowest predictive performance because the observations are collected further from the bankruptcy onset, thus increasing forecasting uncertainty. In addition, the two-year-ahead prediction setting excludes financial information from the year immediately preceding bankruptcy, $t-1$, which typically contains the predictive and informative distress signals, such as sharp deterioration in liquidity, leverage, and profitability ratios. As a result, the models must rely on earlier financial patterns that are less directly indicative of imminent bankruptcy risk. Across feature sets, the data-driven selections, the cumulative-importance and RFECV sets, yield higher AUC values, while the compact five-ratio Altman set yields the superior detection in terms of G-mean at the default threshold for the deep learning models (e.g., BiLSTM). This indicates that a larger, data-driven feature sets improve probability ranking, as captured by AUC, while the classical Altman ratios remain highly competitive for threshold-based bankruptcy detection. In most cases, among the two deep learning models, BiLSTM consistently outperforms CNN, attaining higher AUC and a higher G-mean across feature sets and prediction horizons. Taken together, these results do not identify a single universally best model. Rather, model performance depends on the evaluation criterion and the operating threshold. Tree-based ensembles perform best in terms of AUC, whereas BiLSTM is the best-balanced detector at the conventional threshold in terms of G-mean. From a practical standpoint, the default-threshold behavior is the more relevant one, since a model is normally deployed at a fixed threshold rather than one re-tuned to each new sample, and only the deep models remain usable detectors under that constraint. This motivates the multi-criterion evaluation adopted in this study, in which models are compared not only by AUC but also by recall, specificity, G-mean, type I and type II errors, MCC, and the Brier score. Table 8 and Table 9 report the comprehensive set of performance measures for the benchmark and proposed models across feature sets and prediction horizons, respectively. Table 10 and Table 11 provide a comparison of model performance for the one- and two-year-ahead horizons, respectively, showing how G-mean changes under the conventional threshold, the base-rate threshold, and each model's optimal threshold from the threshold sweep. These results demonstrate that model rankings are threshold-dependent and that conclusions based solely on the default threshold may overlook substantially better prediction models.

Table 8. Performance metrics at the conventional decision threshold (t = 0.5) for the one-year-ahead horizon. Values averaged over 100 random 70/30 company-level splits, with the best result for each metric highlighted in bold.

Feature set	Model	Precision	Recall	Specificity	F1	G-mean	AUC	Type I err	Type II err	MCC	Brier
Altman (5)	BiLSTM	0.061	0.629	0.686	0.110	0.653	0.713	0.314	0.371	0.117	0.197
Altman (5)	CNN	0.051	0.583	0.645	0.094	0.604	0.645	0.355	0.417	0.084	0.229
Altman (5)	LogReg	0.038	0.681	0.454	0.073	0.554	0.593	0.546	0.319	0.047	0.252
Altman (5)	LDA	0.132	0.013	0.997	0.023	0.078	0.548	0.003	0.987	0.030	0.032
Altman (5)	RandomForest	0.030	0.001	1.000	0.001	0.004	0.740	0.000	0.999	0.004	0.029
Altman (5)	XGBoost	0.074	0.459	0.817	0.127	0.610	0.723	0.183	0.541	0.121	0.131
Altman (5)	LightGBM	0.120	0.202	0.953	0.150	0.434	0.721	0.047	0.798	0.120	0.056
Barboza (11)	BiLSTM	0.062	0.526	0.743	0.110	0.617	0.692	0.257	0.474	0.106	0.177
Barboza (11)	CNN	0.039	0.462	0.623	0.071	0.527	0.537	0.377	0.538	0.032	0.253
Barboza (11)	LogReg	0.043	0.446	0.685	0.079	0.549	0.563	0.315	0.554	0.049	0.229
Barboza (11)	LDA	0.068	0.012	0.994	0.020	0.075	0.540	0.006	0.988	0.015	0.034
Barboza (11)	RandomForest	0.123	0.003	1.000	0.006	0.020	0.762	0.000	0.997	0.018	0.032
Barboza (11)	XGBoost	0.157	0.040	0.993	0.062	0.178	0.714	0.007	0.960	0.064	0.033
Barboza (11)	LightGBM	0.128	0.141	0.969	0.133	0.364	0.719	0.031	0.859	0.105	0.047
RFECV	BiLSTM	0.069	0.465	0.786	0.116	0.586	0.704	0.214	0.535	0.108	0.145
RFECV	CNN	0.049	0.481	0.699	0.088	0.571	0.633	0.301	0.519	0.068	0.206
RFECV	LogReg	0.053	0.574	0.669	0.096	0.614	0.693	0.331	0.426	0.089	0.188
RFECV	LDA	0.071	0.022	0.989	0.030	0.114	0.605	0.011	0.978	0.019	0.038
RFECV	RandomForest	0.125	0.006	1.000	0.011	0.031	0.781	0.000	0.994	0.024	0.031
RFECV	XGBoost	0.207	0.053	0.993	0.082	0.213	0.746	0.007	0.947	0.088	0.032
RFECV	LightGBM	0.298	0.014	0.999	0.025	0.078	0.763	0.001	0.986	0.055	0.031
Cumulative-imp.	BiLSTM	0.071	0.337	0.846	0.110	0.515	0.679	0.154	0.663	0.090	0.117
Cumulative-imp.	CNN	0.048	0.405	0.742	0.085	0.541	0.618	0.258	0.595	0.058	0.203
Cumulative-imp.	LogReg	0.057	0.508	0.732	0.102	0.609	0.689	0.268	0.492	0.093	0.166
Cumulative-imp.	LDA	0.064	0.035	0.983	0.044	0.167	0.586	0.017	0.965	0.024	0.043
Cumulative-imp.	RandomForest	0.050	0.001	1.000	0.003	0.008	0.790	0.000	0.999	0.008	0.030
Cumulative-imp.	XGBoost	0.306	0.047	0.997	0.079	0.199	0.762	0.003	0.953	0.107	0.030
Cumulative-imp.	LightGBM	0.298	0.011	1.000	0.022	0.065	0.776	0.000	0.989	0.054	0.030

Table 9. Performance metrics at the conventional decision threshold (t = 0.5) for the two-year-ahead horizon. Values averaged over 100 random 70/30 company-level splits, with the best result for each metric highlighted in bold.

Feature set	Model	Precision	Recall	Specificity	F1	G-mean	AUC	Type I err	Type II err	MCC	Brier
Altman (5)	BiLSTM	0.051	0.600	0.640	0.093	0.615	0.651	0.360	0.400	0.087	0.220
Altman (5)	CNN	0.044	0.547	0.610	0.081	0.566	0.602	0.390	0.453	0.057	0.229
Altman (5)	LogReg	0.039	0.653	0.481	0.073	0.558	0.587	0.519	0.347	0.047	0.251
Altman (5)	LDA	0.031	0.002	0.998	0.004	0.015	0.564	0.002	0.998	0.000	0.032
Altman (5)	RandomForest	0.000	0.000	1.000	0.000	0.000	0.679	0.000	1.000	−0.000	0.030
Altman (5)	XGBoost	0.063	0.412	0.806	0.109	0.574	0.673	0.194	0.588	0.094	0.141
Altman (5)	LightGBM	0.081	0.157	0.943	0.106	0.380	0.671	0.057	0.843	0.073	0.064
Barboza (11)	BiLSTM	0.055	0.501	0.724	0.099	0.597	0.652	0.276	0.499	0.087	0.193
Barboza (11)	CNN	0.037	0.440	0.629	0.067	0.515	0.524	0.371	0.560	0.025	0.243
Barboza (11)	LogReg	0.037	0.419	0.649	0.068	0.516	0.526	0.351	0.581	0.025	0.236
Barboza (11)	LDA	0.021	0.003	0.995	0.005	0.019	0.549	0.005	0.997	−0.004	0.034
Barboza (11)	RandomForest	0.010	0.000	1.000	0.000	0.001	0.710	0.000	1.000	0.001	0.033
Barboza (11)	XGBoost	0.131	0.037	0.992	0.056	0.176	0.656	0.008	0.963	0.053	0.035
Barboza (11)	LightGBM	0.091	0.112	0.965	0.100	0.320	0.665	0.035	0.888	0.069	0.053
RFECV	BiLSTM	0.053	0.425	0.748	0.091	0.540	0.647	0.252	0.575	0.070	0.162
RFECV	CNN	0.045	0.462	0.690	0.082	0.553	0.614	0.310	0.538	0.057	0.200
RFECV	LogReg	0.045	0.595	0.601	0.084	0.592	0.663	0.399	0.405	0.070	0.204
RFECV	LDA	0.029	0.005	0.993	0.007	0.029	0.611	0.007	0.995	−0.003	0.035
RFECV	RandomForest	0.045	0.001	1.000	0.003	0.008	0.714	0.000	0.999	0.007	0.032
RFECV	XGBoost	0.146	0.036	0.992	0.055	0.163	0.682	0.008	0.964	0.055	0.034
RFECV	LightGBM	0.143	0.006	0.999	0.011	0.036	0.693	0.001	0.994	0.023	0.031
Cumulative-imp.	BiLSTM	0.056	0.289	0.835	0.088	0.470	0.630	0.165	0.711	0.059	0.125
Cumulative-imp.	CNN	0.047	0.401	0.742	0.084	0.538	0.611	0.258	0.599	0.056	0.195
Cumulative-imp.	LogReg	0.047	0.517	0.667	0.086	0.585	0.646	0.333	0.483	0.067	0.189
Cumulative-imp.	LDA	0.026	0.009	0.987	0.013	0.055	0.576	0.013	0.991	−0.004	0.040
Cumulative-imp.	RandomForest	0.000	0.000	1.000	0.000	0.000	0.727	0.000	1.000	−0.000	0.031
Cumulative-imp.	XGBoost	0.172	0.023	0.997	0.041	0.125	0.697	0.003	0.977	0.052	0.031
Cumulative-imp.	LightGBM	0.082	0.003	1.000	0.006	0.021	0.710	0.000	0.997	0.013	0.031

Table 10. Summary of the G-mean results for the one-year-ahead prediction horizon at the conventional threshold (t = 0.5), at the base-rate threshold (t = 0.031), and each model's optimal threshold from the threshold sweep. The results show that the same models

Feature set	Model	G-mean at t = 0.5	G-mean at t = 0.031	Highest G-mean (at threshold)
Altman (5)	BiLSTM	0.653	0.095	0.653 (t = 0.5)
Altman (5)	CNN	0.604	0.056	0.604 (t = 0.5)
Altman (5)	LogReg	0.554	0.022	0.554 (t = 0.5)
Altman (5)	LDA	0.078	0.408	0.484 (t = 0.02)
Altman (5)	RandomForest	0.004	0.666	0.671 (t = 0.05)
Altman (5)	XGBoost	0.610	0.284	0.646 (t = 0.4)
Altman (5)	LightGBM	0.434	0.606	0.658 (t = 0.075)
Barboza (11)	BiLSTM	0.617	0.102	0.617 (t = 0.5)
Barboza (11)	CNN	0.527	0.322	0.527 (t = 0.5)
Barboza (11)	LogReg	0.549	0.152	0.549 (t = 0.5)
Barboza (11)	LDA	0.075	0.377	0.501 (t = 0.02)
Barboza (11)	RandomForest	0.020	0.566	0.692 (t = 0.075)
Barboza (11)	XGBoost	0.178	0.636	0.652 (t = 0.02)
Barboza (11)	LightGBM	0.364	0.634	0.658 (t = 0.05)
RFECV	BiLSTM	0.586	0.326	0.603 (t = 0.4)
RFECV	CNN	0.571	0.545	0.573 (t = 0.4)
RFECV	LogReg	0.614	0.462	0.614 (t = 0.5)
RFECV	LDA	0.114	0.380	0.465 (t = 0.02)
RFECV	RandomForest	0.031	0.605	0.708 (t = 0.075)
RFECV	XGBoost	0.213	0.557	0.635 (t = 0.01)
RFECV	LightGBM	0.078	0.224	0.277 (t = 0.01)
Cumulative-imp.	BiLSTM	0.515	0.357	0.568 (t = 0.3)
Cumulative-imp.	CNN	0.541	0.572	0.575 (t = 0.1)
Cumulative-imp.	LogReg	0.609	0.544	0.624 (t = 0.4)
Cumulative-imp.	LDA	0.167	0.344	0.525 (t = 0.01)
Cumulative-imp.	RandomForest	0.008	0.603	0.712 (t = 0.075)
Cumulative-imp.	XGBoost	0.199	0.527	0.626 (t = 0.01)
Cumulative-imp.	LightGBM	0.065	0.214	0.263 (t = 0.01)

Table 11. Summary of the G-mean results for the two-year-ahead prediction horizon at the conventional threshold (t = 0.5), at the base-rate threshold (t = 0.031), and each model's optimal threshold from the threshold sweep. The results show that the same models can rank differently under various amounts of threshold.

Feature set	Model	G-mean at t = 0.5	G-mean at t = 0.031	Highest G-mean (at threshold)
Altman (5)	BiLSTM	0.615	0.036	0.615 (t = 0.5)
Altman (5)	CNN	0.566	0.065	0.566 (t = 0.5)
Altman (5)	LogReg	0.558	0.019	0.558 (t = 0.5)
Altman (5)	LDA	0.015	0.452	0.452 (t = 0.031)
Altman (5)	RandomForest	0.000	0.615	0.626 (t = 0.05)
Altman (5)	XGBoost	0.574	0.107	0.623 (t = 0.4)
Altman (5)	LightGBM	0.380	0.526	0.628 (t = 0.1)
Barboza (11)	BiLSTM	0.597	0.048	0.597 (t = 0.5)
Barboza (11)	CNN	0.515	0.314	0.515 (t = 0.5)
Barboza (11)	LogReg	0.516	0.118	0.516 (t = 0.5)
Barboza (11)	LDA	0.019	0.401	0.401 (t = 0.031)
Barboza (11)	RandomForest	0.001	0.505	0.646 (t = 0.075)
Barboza (11)	XGBoost	0.176	0.603	0.609 (t = 0.02)
Barboza (11)	LightGBM	0.320	0.568	0.621 (t = 0.075)
RFECV	BiLSTM	0.540	0.267	0.562 (t = 0.4)
RFECV	CNN	0.553	0.522	0.559 (t = 0.4)
RFECV	LogReg	0.592	0.413	0.592 (t = 0.5)
RFECV	LDA	0.029	0.436	0.436 (t = 0.031)
RFECV	RandomForest	0.008	0.535	0.651 (t = 0.075)
RFECV	XGBoost	0.163	0.522	0.598 (t = 0.01)
RFECV	LightGBM	0.036	0.150	0.215 (t = 0.01)
Cumulative-imp.	BiLSTM	0.470	0.280	0.528 (t = 0.3)
Cumulative-imp.	CNN	0.538	0.565	0.568 (t = 0.075)
Cumulative-imp.	LogReg	0.585	0.479	0.590 (t = 0.4)
Cumulative-imp.	LDA	0.055	0.360	0.533 (t = 0.02)
Cumulative-imp.	RandomForest	0.000	0.532	0.659 (t = 0.075)
Cumulative-imp.	XGBoost	0.125	0.490	0.585 (t = 0.01)
Cumulative-imp.	LightGBM	0.021	0.123	0.189 (t = 0.01)

Table 12. Paired Wilcoxon signed-rank tests comparing BiLSTM against each benchmark model in terms of AUC and G-mean at t = 0.5, based on 100 random 70/30 company-level splits. Positive Δ values indicate higher performance for BiLSTM. Significance levels are denoted as follows: ***: $p < 0.001$, **: $p < 0.01$, *: $p < 0.05$, and ns: $p \geq 0.05$.

Horizon	Feature set	Benchmark	ΔAUC	P-value (AUC)	Sig.	ΔG-mean	P-value (G-mean)	Sig.
1-year	Altman (5)	CNN	0.068	2.4e-15	***	0.049	1.1e-11	***
1-year	Altman (5)	LogReg	0.120	5.1e-18	***	0.099	5.3e-18	***
1-year	Altman (5)	LDA	0.166	4.1e-18	***	0.576	3.9e-18	***
1-year	Altman (5)	RandomForest	−0.026	4.9e-10	***	0.649	3.9e-18	***
1-year	Altman (5)	XGBoost	−0.009	0.016	*	0.043	6.1e-12	***
1-year	Altman (5)	LightGBM	−0.007	0.115	ns	0.220	3.9e-18	***
1-year	Barboza (11)	CNN	0.156	4.8e-18	***	0.090	1.3e-14	***
1-year	Barboza (11)	LogReg	0.130	4e-18	***	0.069	2.5e-13	***
1-year	Barboza (11)	LDA	0.152	4.3e-18	***	0.543	3.9e-18	***
1-year	Barboza (11)	RandomForest	−0.070	5.1e-18	***	0.597	3.9e-18	***
1-year	Barboza (11)	XGBoost	−0.022	1.9e-06	***	0.439	3.9e-18	***
1-year	Barboza (11)	LightGBM	−0.027	1.1e-08	***	0.254	3.9e-18	***
1-year	RFECV	CNN	0.071	9.8e-17	***	0.015	0.012	*
1-year	RFECV	LogReg	0.011	0.041	*	−0.028	0.022	*
1-year	RFECV	LDA	0.099	5.9e-18	***	0.473	3.9e-18	***
1-year	RFECV	RandomForest	−0.078	7.3e-18	***	0.555	3.9e-18	***
1-year	RFECV	XGBoost	−0.042	1.2e-10	***	0.373	3.9e-18	***
1-year	RFECV	LightGBM	−0.060	2.5e-15	***	0.508	3.9e-18	***
1-year	Cumulative-imp.	CNN	0.062	3.8e-15	***	−0.026	0.098	ns
1-year	Cumulative-imp.	LogReg	−0.010	0.123	ns	−0.093	1.2e-11	***
1-year	Cumulative-imp.	LDA	0.093	9.6e-18	***	0.348	3.9e-18	***
1-year	Cumulative-imp.	RandomForest	−0.111	3.9e-18	***	0.507	3.9e-18	***
1-year	Cumulative-imp.	XGBoost	−0.082	4.1e-18	***	0.317	3.9e-18	***
1-year	Cumulative-imp.	LightGBM	−0.096	3.9e-18	***	0.450	3.9e-18	***
2-year	Altman (5)	CNN	0.048	2.3e-13	***	0.049	3.3e-12	***
2-year	Altman (5)	LogReg	0.064	5e-17	***	0.057	8.7e-15	***
2-year	Altman (5)	LDA	0.087	5.4e-18	***	0.600	3.9e-18	***
2-year	Altman (5)	RandomForest	−0.028	2.9e-10	***	0.615	3.9e-18	***
2-year	Altman (5)	XGBoost	−0.022	2.6e-07	***	0.041	6.3e-10	***
2-year	Altman (5)	LightGBM	−0.020	7e-06	***	0.235	3.9e-18	***
2-year	Barboza (11)	CNN	0.128	4.1e-18	***	0.082	3.3e-14	***
2-year	Barboza (11)	LogReg	0.125	5.3e-18	***	0.082	2.6e-16	***
2-year	Barboza (11)	LDA	0.102	1.5e-17	***	0.578	3.9e-18	***
2-year	Barboza (11)	RandomForest	−0.058	5.8e-17	***	0.596	3.9e-18	***
2-year	Barboza (11)	XGBoost	−0.004	0.698	ns	0.422	3.9e-18	***
2-year	Barboza (11)	LightGBM	−0.013	0.032	*	0.277	3.9e-18	***
2-year	RFECV	CNN	0.033	3.2e-09	***	−0.013	0.611	ns
2-year	RFECV	LogReg	−0.016	0.00032	***	−0.052	4.6e-06	***

Continued on next page

Horizon	Feature set	Benchmark	ΔAUC	P-value (AUC)	Sig.	ΔG-mean	P-value (G-mean)	Sig.
2-year	RFECV	LDA	0.036	2.1e-08	***	0.511	3.9e-18	***
2-year	RFECV	RandomForest	−0.067	2.8e-17	***	0.532	3.9e-18	***
2-year	RFECV	XGBoost	−0.035	7.9e-10	***	0.377	3.9e-18	***
2-year	RFECV	LightGBM	−0.046	4e-12	***	0.504	3.9e-18	***
2-year	Cumulative-imp.	CNN	0.019	0.00056	***	−0.068	1.6e-06	***
2-year	Cumulative-imp.	LogReg	−0.017	0.003	**	−0.115	4.2e-15	***
2-year	Cumulative-imp.	LDA	0.054	4.3e-13	***	0.415	3.9e-18	***
2-year	Cumulative-imp.	RandomForest	−0.097	3.9e-18	***	0.470	3.9e-18	***
2-year	Cumulative-imp.	XGBoost	−0.067	2.5e-17	***	0.345	4.5e-18	***
2-year	Cumulative-imp.	LightGBM	−0.080	1.1e-17	***	0.449	3.9e-18	***

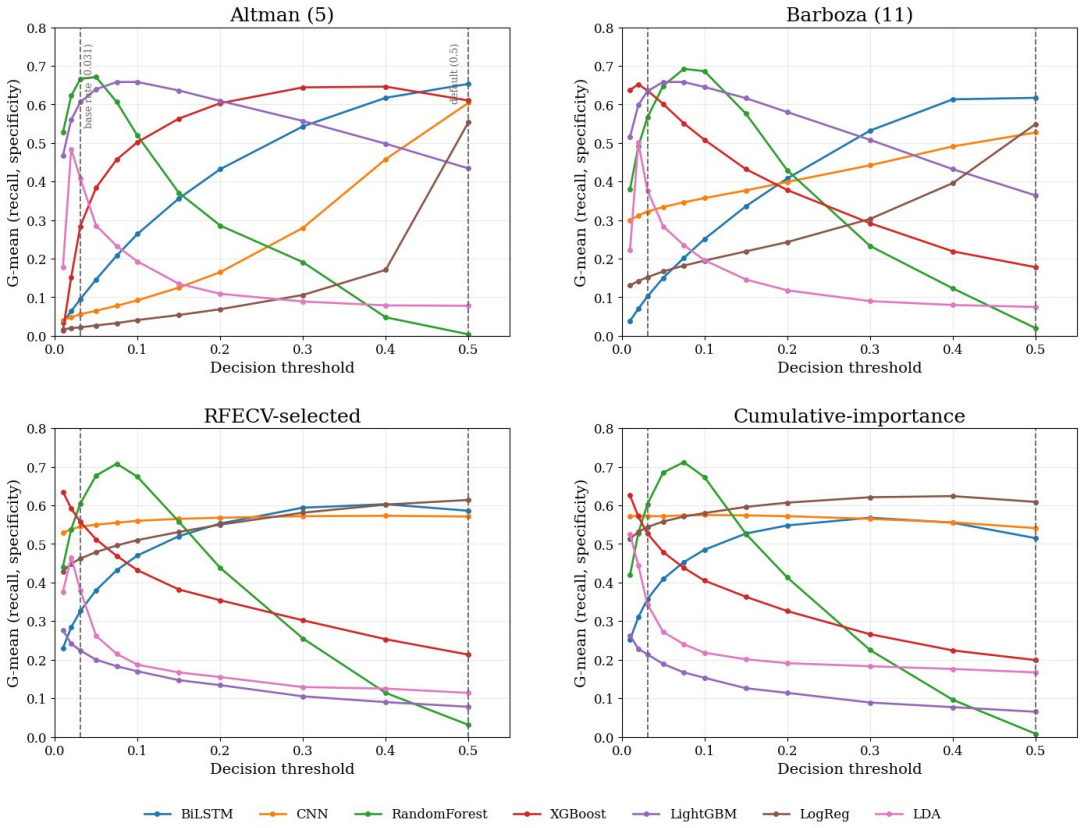

Figure 7. Comparison of the G-mean versus decision threshold for the one-year-ahead horizon, shown separately for each feature set: Altman, Barboza, RFECV-selected, and cumulative-importance. Each curve represents a model, and the dashed vertical lines indicate the bankruptcy base-rate threshold (t = 0.031) and the conventional default threshold (t = 0.5).

Figure 7 shows that the threshold-dependent behavior of the models is broadly consistent across the four feature sets. The tree-based ensembles, particularly random forest and XGBoost, generally

achieve their highest G-mean at relatively low thresholds near the bankruptcy base rate t = 0.031 and then decline as the threshold approaches the conventional value of t = 0.5. This pattern is especially pronounced for random forest, whose balanced detection deteriorates sharply at higher thresholds. Under severe class imbalance, predicted bankruptcy probabilities often remain well below 0.5, so the threshold of 0.5 suppresses many positive predictions and reduces recall. By contrast, BiLSTM maintains relatively high G-mean values at higher thresholds and is one of the most effective detectors at the conventional operating point (t = 0.5). CNN shows a similar but less consistent pattern. Logistic regression behaves differently from the tree ensembles in several panels, often improving as the threshold increases. Overall, the figure demonstrates that model rankings are highly threshold-dependent: Tree-based ensembles can achieve high peak performance when the threshold is tuned toward the base rate, whereas BiLSTM remains comparatively robust at t = 0.5. The recurrence of this pattern across the Altman, Barboza, RFECV-selected, and cumulative-importance feature sets suggests that the contrast is driven more by model behavior under class imbalance than by the specific feature set used.

The dependence of model ranking on the decision threshold is consistent with a well-established literature on cost-sensitive classification, in which the conventional threshold of 0.5 is shown to be optimal only when the two classes are balanced and the two types of misclassifications carry the same cost [76–78]. Neither condition holds in bankruptcy prediction: The sample is severely imbalanced and the cost of failing to flag a company that subsequently fails (a type II error) is usually much larger than the cost of a false alarm [8,79]. Tree-based models with well-calibrated probabilities concentrate their predicted scores well below 0.5 under such imbalance, so applying the default threshold effectively suppresses all positive predictions and produces the near-zero recall observed in Tables 8 and 9. When the threshold is allowed to vary, the same random forest models recover and reach the highest geometric means in our experiments (Table 10 for the one-year-ahead horizon and Table 11 for the two-year-ahead horizon), with peaks near the bankruptcy base rate ($t \approx 0.075$). This pattern, summarized in Figure 7, motivates evaluating models across a range of thresholds rather than at a single fixed value, particularly when the operating cost ratio is uncertain or stakeholder dependent.

The trade-off between the classical and data-driven feature sets is also informative. The compact five-ratio Altman set yields the best balanced detection at the conventional threshold for the deep learning models, while the larger data-driven sets, RFECV and cumulative importance, yield the highest AUC values and the highest G-means for the tree ensembles, as shown in Table 10. This indicates that the Altman ratios encode information that is discriminative at the default operating point (t = 0.5), but that larger feature sets are required to maximize probability ranking performance.

Taken together, these results suggest that the choice of the best model for bankruptcy prediction is inseparable from the user's loss function. A lender who places greater weight on missed defaults than on false alarms may prefer to operate random forest at a low threshold, around $t \approx 0.05$–0.075, where it achieves a G-mean above 0.70 with the data-driven feature sets. By contrast, a regulator or auditor who assigns approximately equal costs to missed defaults and false alarms may prefer BiLSTM with the Altman ratios at the conventional threshold, as it provides a balanced detection profile without requiring threshold tuning. The full set of performance measures and threshold sweeps reported here is intended to support both types of decision-making.

To determine whether the performance differences between models are statistically significant, BiLSTM is further compared against each benchmark model using paired Wilcoxon signed-rank tests based on the 100 paired AUC and G-mean values for each model comparison, as shown in Table 12. The results confirm the criterion-dependent pattern observed above. In terms of AUC, BiLSTM performs significantly better than CNN, logistic regression, LDA, random forest, XGBoost, and LightGBM in most cases. However, it performs significantly worse compared to logistic regression with the cumulative-importance feature set and LightGBM with the Altman five-feature set for the one-year-ahead prediction horizon, as well as XGBoost with the Barboza 11-feature set for the two-year-ahead prediction horizon. In contrast, for the G-mean at the threshold 0.5, BiLSTM significantly outperforms every benchmark model except CNN with the cumulative-importance feature set for the one-year-ahead horizon and CNN with the RFECV feature set for the two-year-ahead horizon. The differences are especially large relative to the tree-based ensemble models, whose balanced detection deteriorates at the threshold of 0.5. Nearly all comparisons are statistically significant at the 0.1% level.

To assess whether the reported performance is robust to the macroeconomic conditions and industry composition represented in the 1994 to 2020 sample, we conduct a stratified robustness analysis. Specifically, we pool the one-year-ahead out-of-sample predictions across all 100 random seeds, stratify them by economic sub-period and SIC industry division, and recompute the AUC within each stratum. In Table 13 and Table 14, n refers to the number of pooled test predictions in each stratum and "events" is the number of predictions that corresponding to bankrupt companies. The AUC is computed once using the pooled predictions, rather than averaged across seeds, to ensure that even relatively sparse strata contain a sufficient number of bankruptcy events for estimation. For brevity, we report results for the strongest ranking-based model, random forest with the cumulative-importance feature set, and the proposed deep learning model, BiLSTM, with the Altman feature set.

Table 13. One-year-ahead AUC by economic sub-period, computed on out-of-sample predictions pooled across the 100 seeds, where *n* refers to the number of pooled predictions and events the number corresponding to bankrupt companies.

Period	n	Events	RF + cumimp	BiLSTM + Altman
≤ 2001 (incl. dot-com)	40,645	2,141	0.751	0.715
2002–2007	29,502	1,310	0.751	0.669
2008–2012 (GFC)	18,431	893	0.752	0.687
2013–2020	58,022	156	0.820	0.668

Model discrimination is generally stable across the major economic periods in which bankruptcy events are sufficiently represented. For the random forest model, the AUC remains essentially unchanged across the pre-2002 period, the 2002–2007 expansion period, and the 2008–2012 financial-crisis window, with values of 0.751, 0.751, and 0.752, respectively. This indicates no deterioration in discriminatory performance during the financial-crisis period. The BiLSTM shows a similar pattern, with AUC values of 0.715, 0.669, and 0.687 across the same periods. The 2013–2020 window should be interpreted with caution because its estimates primarily reflect the structure of the panel and right-censoring rather than a clear change in model performance. Each company is assigned to the period corresponding to its last year in the panel. For bankrupt companies, this year corresponds to the

bankruptcy year; however, for surviving companies, it corresponds only to the final year in which the company appears in the data. Because many surviving companies remain in the panel through the 2020 sample cutoff, the 2013–2020 window consists almost entirely of surviving companies, with approximately 58,000 pooled predictions and only a small number of distinct bankruptcy events. Its failure rate is therefore approximately 0.3%, compared with 4–5% in earlier periods. As a result, the AUC for this most recent window is estimated from too few events to be considered reliable. Importantly, the bankruptcy events in the sample are concentrated in 1994–2012, a period that includes the dot-com downturn and the financial crisis.

Table 14. One-year-ahead AUC by SIC industry division, computed on out-of-sample predictions pooled across the 100 seeds. No AUC is reported for the construction division because it contains no bankruptcy events in the sample.

Division	n	Events	RF + cumimp	BiLSTM + Altman
Manufacturing	90,099	2,556	0.809	0.716
Services	38,330	1,371	0.723	0.643
Transportation & Utilities	11,443	515	0.817	0.796
Wholesale Trade	5,592	58	0.786	0.744
Construction	1,136	0	—	—

A similar pattern of stability is observed across industries. In the three SIC divisions that account for most of the sample, *Manufacturing, Services,* and *Transportation & Utilities*, the random forest achieves AUC values of 0.81, 0.72, and 0.82, respectively, while the BiLSTM achieves AUC values of 0.72, 0.64, and 0.80, respectively. The *Wholesale Trade* division contains too few bankruptcy events, and the Construction division contains none, so the corresponding estimates should be interpreted with caution or are not estimable. Because the sample is concentrated primarily in *Manufacturing* and *Services*, which together account for the large majority of companies, we note this industry composition as a scope condition of the study. Taken together, the stratified results show no systematic decline in discrimination across economic regimes, including the global financial crisis, or across the well-populated industry sectors. This indicates that the predictive performance of the models reflects company-level financial fundamentals that remain informative across the economic and industry conditions represented in the sample, rather than being driven by a particular time period or industry. Nevertheless, a formal out-of-time evaluation and the explicit inclusion of macroeconomic variables, year effects, or industry-level covariates remain important extensions of this analysis, as discussed in Section 5.

4.4. Model interpretations

In this section, we interpret the configuration that achieves the highest discrimination in Table 10 of Section 4.3: The random forest classifier applies to the two data-driven feature sets, cumulative-importance and RFECV, at a threshold of 0.075. We use Shapley Additive Explanations (SHAP) [80] to analyze the effect of these predictor variables. As Section 4.3 shows, model superiority is criterion-dependent. The tree ensembles, led by random forest, achieve the highest AUC values, 0.790 and 0.781

for the cumulative-importance and RFECV sets, respectively. They also result in the highest peak G-means, with values of 0.712 and 0.708 for the cumulative-importance and RFECV feature sets, respectively. By contrast, BiLSTM provides the strongest balanced detection at the conventional decision threshold. We focus the interpretation on random forest for two reasons. First, it leads to the threshold-independent criteria considered here. Second, as a tree ensemble, it enables exact Shapley-value attribution through TreeExplainer, avoiding the sampling approximations required for sequence models. Because the data-driven feature sets are shared across all models, the resulting importance rankings characterize the feature space used throughout the study. SHAP values quantify each feature's marginal contribution to a prediction using a cooperative game-theoretic approach, and we aggregate them across companies and time steps to obtain a single mean-absolute SHAP value for each variable. To assist economic interpretation, the SHAP plots use self-explanatory variable labels, such as "Total liabilities" instead of "lt" and "Working capital/Total assets" instead of "A".

Figures 8 and 9 report the fifteen most influential variables for the random forest using the cumulative-importance and RFECV feature sets, respectively. The horizontal axis measures the mean absolute SHAP value in units of predicted bankruptcy probability. Each value represents the average magnitude across all companies and time steps, by which a given variable shifts the predicted probability of default away from the model's baseline prediction. Because the unconditional bankruptcy rate in the sample is low (approximately 3%), the predicted probabilities and the corresponding SHAP contributions are small in absolute terms. Accordingly, the plots should be interpreted by comparing the relative magnitudes of the bars rather than their absolute value.

The two feature sets produce a consistent interpretation. In both cases, the three most influential variables are the Altman ratios: EBIT/total assets, retained earnings/total assets, and working capital/total assets. These are followed by income before extraordinary items, the two Altman composite scores, the original Z-score and the four-variable Z"-score, and market-based variables such as the fiscal year-end share price and earnings per share.

Fourteen of the fifteen highest-ranked variables overlap across the two selection methods, indicating that the SHAP attributions capture genuine economic signal rather than the idiosyncrasies of a particular selection rule. Notably, even within the much larger data-driven feature sets, containing 56 and 68 variables, respectively, the classic Altman solvency and profitability ratios dominate the model's risk assessment. The Altman composite scores also rank among the leading contributors. Thus, the SHAP analysis confirms the enduring relevance of the Altman framework, while showing that the data-driven feature sets add incremental information through variables such as common-stock capital, asset growth, and operating margin. These rankings also align with the selection-frequency analysis presented in Table 6: the variables on which the model relies most are also those most consistently selected across the random splits. Feature-importance and selection-stability evidence therefore converge on the same small set of core predictors.

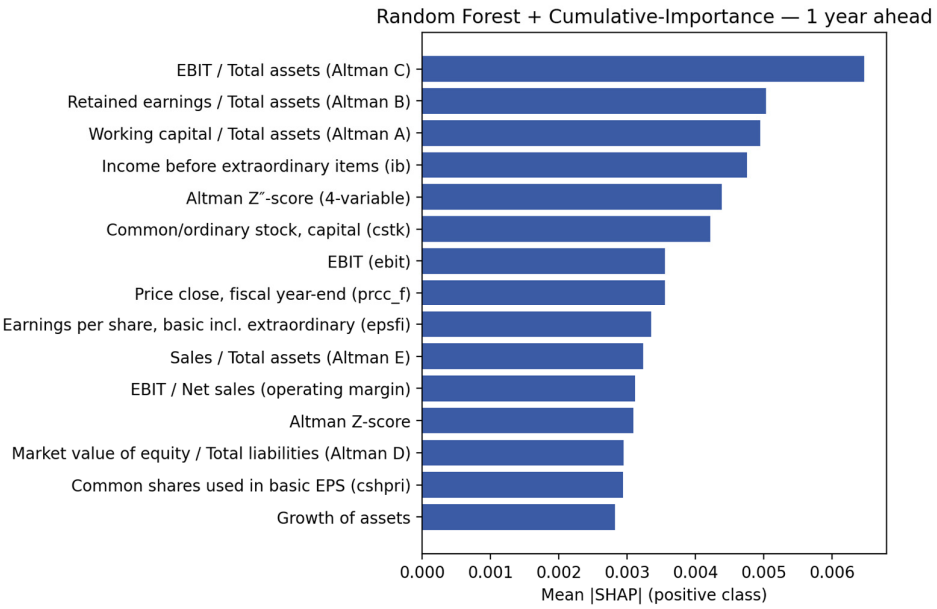

Figure 8. The fifteen most influential variables, measured by mean absolute SHAP, for the random forest with the cumulative-importance feature set at the one-year-ahead horizon.

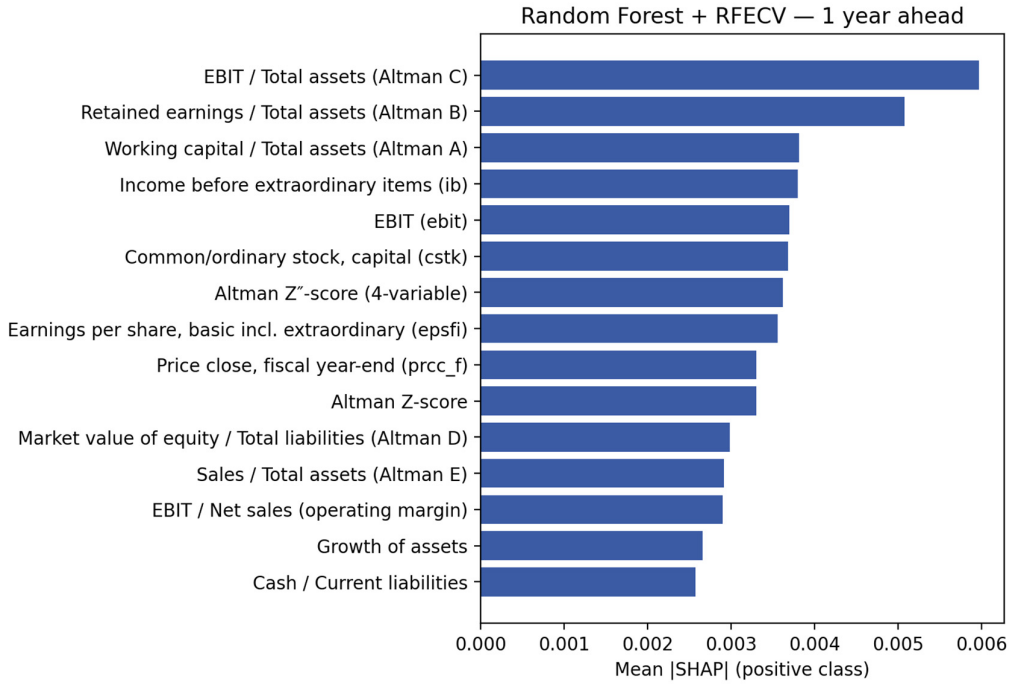

Figure 9. The fifteen most influential variables, measured by mean absolute SHAP, for the random forest with the RFECV feature set at the one-year-ahead horizon.

To check that the proposed deep model bases its predictions on economically sensible variables rather than spurious patterns, we also apply SHAP to the best-performing BiLSTM configuration. Because the BiLSTM achieves its strongest balanced detection performance using the Altman feature set, we focus the interpretation on this specification. We use a gradient-based SHAP explainer appropriate for recurrent neural networks and compute feature importance by averaging the absolute SHAP values across companies and time steps.

Figure 10 presents the resulting feature-importance ranking. The BiLSTM relies most heavily on EBIT/total assets (Altman C) and retained earnings/total assets (Altman B), while the market-leverage, liquidity, and turnover ratios make smaller contributions. This ranking is highly consistent with the ordering obtained from the random forest model and aligns with the established role of operating profitability and accumulated retained earnings in bankruptcy and financial-distress prediction. The fact that a tree-based ensemble and a bidirectional recurrent neural network identify the same leading predictors suggests that the main predictive signal is driven by interpretable financial fundamentals, rather than the behavior of any one model.

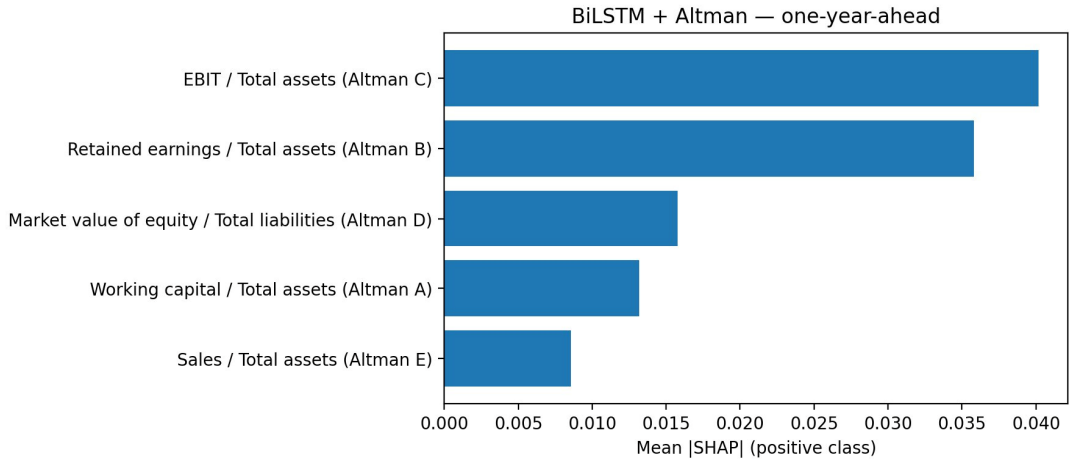

Figure 10. Variable importance, measured by mean absolute SHAP, for the BiLSTM model with the Altman feature set at the one-year-ahead prediction horizon.

5. Concluding remarks

Financial risk analysis is a significant aspect to many financial and business institutions and its beneficiaries. It is necessary that any institution should avoid a situation of poor financial status, such as bankruptcy or liquidation of assets to protect its beneficiaries. Thus, the study of financial risk analysis has been focused on for a long time and continuous to be so. In our experiment, we attempt to predict the status of companies using various financial features. Our computational results demonstrate that temporal data helps in efficiently capturing the behavior of companies using advanced machine learning algorithms. In particular, we examined the effectiveness of two well-known deep learning algorithms for the bankruptcy prediction problem. We showed that our BiLSTM would improve predictive accuracy using past and future time steps. Moreover, we used the recursive feature

elimination technique to select important features from a large set of features and compared them with traditional financial indicators.

Our findings of this study have important practical implications for financial institutions, investors, and policymakers. Accurate early bankruptcy prediction models can assist banks, lenders, and credit agencies in improving credit risk assessment, loan approval processes, portfolio management, and financial monitoring systems. In addition, investors and business analysts may use such predictive tools to better evaluate corporate financial stability and investment risk. From a regulatory perspective, the proposed framework may help policymakers and financial oversight agencies identify financially vulnerable companies earlier and enable more proactive intervention strategies and improved monitoring of systemic financial risk during periods of economic uncertainty.

We observed that the data-driven feature sets achieve the highest AUC values, while the compact Altman five-ratio set is the most effective for balanced detection at the conventional threshold. To show the advantages of our BiLSTM and CNN methods, we performed an extensive experimental study under one-year-ahead and two-year-ahead bankruptcy prediction settings with various sets of features. The comprehensive comparison shows that no single model is best across all criteria. The tree-based ensemble models, led by random forest, achieve the highest AUC, whereas BiLSTM is the strongest balanced detector at the conventional decision threshold and consistently outperforms CNN. Because a model with high AUC may nonetheless fail to flag any bankrupt companies at the default threshold under severe class imbalance, we argue that bankruptcy-prediction models should be assessed using multiple complementary criteria rather than AUC alone. This suggests that BiLSTM is a better prediction model in comparison with the CNN model for the data and features used in this experiment. More substantively, the gap in default-threshold performance highlights the practical contribution of the proposed deep learning models. In deployment, the BiLSTM can identify failing companies without the case-specific threshold tuning required by tree-based ensembles before they detect any bankrupt companies. Moreover, the SHAP analysis indicates that the BiLSTM's predictions are driven by the same economically meaningful Altman ratios that influence the tree-based models. Thus, in this setting, deep learning provides a bankruptcy prediction tool that is reliable at a realistic operating threshold and interpretable.

To discover the effectiveness of the models for two-year-ahead bankruptcy prediction, we conducted experiments using a dataset that excludes the most recent pre-bankruptcy financial year. As expected, the predictive performance in the two-year-ahead setting is lower than that of the one-year-ahead prediction setting, which reveals the increased difficulty of forecasting bankruptcy further in advance when the most informative recent financial information is unavailable.

There are many aspects in which this work can be further explored. One of the extensions would be to explore the quarterly and monthly financial datasets to construct longer time series and further enhance the temporal modeling capability of BiLSTM architectures. Future work will also entail time-based out-of-sample evaluation strategies, including chronological and rolling-window testing, using larger and more comprehensive longitudinal company-level datasets to better reflect real-world bankruptcy prediction settings. Textual data can also be used for deep learning models to predict the financial stability of companies [41]. More advanced feature selection techniques can be implemented with the help of deep learning models [81,82] rather than the static recursive feature elimination technique. An interesting advantage of having the probability scores for the prediction is that the cutoff

level can be customized to analyze the risk of a company. For example, if a company's probability of being bankrupt is 0.3, this can be used to flag the company as risky or under financial distress. Additionally, researchers can incorporate bankruptcy prediction into a supplier selection and portfolio optimization framework, as well as into a supply chain network reliability analysis. A limitation of this study is that the proposed framework does not explicitly account for macroeconomic cycles or industry-specific effects, which may influence bankruptcy risk across time periods and sectors. Future research will entail incorporating temporal economic indicators, industry-level variables, year dummy variables, and industry-adjusted variables to improve model robustness, generalizability, and sensitivity to dynamic economic conditions. We also used company-level to preserve company-specific temporal financial patterns, which are key to the BiLSTM modeling framework; however, industry-level imputation strategies may also be explored in future work to account for sector-specific financial characteristics.

Author contributions

Proposal & original idea, K.B. and T.R.; conceptualization, K.B. and T.R.; modeling, G.R., K.B., and T.R.; methodology, G.R. and T.R.; validation, G.R.; software, G.R.; literature review, G.R., K.B., and T.R.; economic & business implication, K.B.; writing—original draft preparation, G.R..; writing—review & editing, G.R., K.B., and T.R; discussion, G.R., K.B., and T.R.; project administration, K.B. and T.R. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by the Defense Advanced Research Projects Agency (DARPA).

Data availability

Upon acceptance of this paper, we are committed to disseminate our source codes through online repository such as GitHub. The financial data used are from CompStat, which may require subscription fees.

Conflict of interest

There is no conflict of interest regarding the publication of this paper.

Use of Generative-AI Tools Declaration

During the preparation of this work, the author(s) used Claude (Anthropic) to (i) check and improve the grammar and language clarity of the manuscript text, (ii) assist with writing and debugging code used in the analysis, following the authors' own conceptual design and brainstorming of the modeling approach, and (iii) generate plotting code used to produce the figures in this manuscript. All AI-assisted outputs were reviewed, verified, and edited by the author(s), who take full responsibility

for the content, correctness, and originality of the published article, including all code, figures, tables, and their numbering/cross-references.

Acknowledgement

This research was developed with funding from the Defense Advanced Research Projects Agency (DARPA). The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

References

1. T. Telford, S. Mufson, PG&E, the nation's biggest utility company, files for bankruptcy after California wildfires, *Washington Post*, 2019.
2. J. H. Bliss, *Financial and Operating Ratios in Management*, New York: The Ronald Press, 1923.
3. A. C. Littleton, The 2 to 1 ratio analyzed, *Certified Public Accountant*, (1926), 244–246.
4. P. J. Fitzpatrick, *A Comparison of the Ratios of Successful Industrial Enterprises with Those of Failed Companies*, 1932.
5. A. H. Winakor, R. F. Smith, Changes in the financial structure of unsuccessful industrial corporations, *Univ. Ill. Bull.*, 51 (1935).
6. C. Mervin, *Financing Small Corporations: In Five Manufacturing Industries, 1926–36*, National Bureau of Economic Research, 1942.
7. W. H. Beaver, Financial ratios as predictors of failure, *J. Account. Res.*, 4 (1966), 71–111. https://doi.org/10.2307/24901717
8. E. Altman, Financial ratios, discriminant analysis and the prediction of corporate bankruptcy, *J. Finance*, **23** (1968), 589–609. https://doi.org/10.1111/j.1540-6261.1968.tb00843.x
9. E. I. Altman, R. G. Haldeman, P. K. Narayanan, ZETA analysis: A new model to identify bankruptcy risk of corporations, *J. Banking Finance*, **1** (1977), 29–54. https://doi.org/10.1016/0378-4266(77)90017-6
10. E. I. Altman, M. Iwanicz-Drozdowska, E. K. Laitinen, A. Suvas, Financial distress prediction in an international context: A review and empirical analysis of Altman's Z-score model, *J. Int. Financ. Manage. Account.*, 28 (2017), 131–171. https://doi.org/10.1111/jifm.12053
11. F. Barboza, H. Kimura, E. Altman, Machine learning models and bankruptcy prediction, *Expert Syst. Appl.*, **83** (2017), 405–417. https://doi.org/10.1016/j.eswa.2017.04.006
12. R. C. Carton, C. W. Hofer, *Measuring Organizational Performance: Metrics for Entrepreneurship and Strategic Management Research*, Edward Elgar, 2006.
13. L. Cleofas-Sánchez, V. García, A. I. Marqués, J. S. Sánchez, Financial distress prediction using the hybrid associative memory with translation, *Appl. Soft Comput.*, **44** (2016), 144–152. https://doi.org/10.1016/j.asoc.2016.04.005
14. J. Heo, J. Y. Yang, AdaBoost based bankruptcy forecasting of Korean construction companies, *Appl. Soft Comput.*, **24** (2014), 494–499. https://doi.org/10.1016/j.asoc.2014.08.009

15. F. J. López Iturriaga, I. P. Sanz, Bankruptcy visualization and prediction using neural networks: A study of U.S. commercial banks, *Expert Syst. Appl.*, **42** (2015), 2857–2869. https://doi.org/10.1016/j.eswa.2014.11.025
16. C. F. Tsai, Y. F. Hsu, D. C. Yen, A comparative study of classifier ensembles for bankruptcy prediction, *Appl. Soft Comput.*, **24** (2014), 977–984. https://doi.org/10.1016/j.asoc.2014.08.047
17. G. P. Naidu, K. Govinda, Bankruptcy prediction using neural networks, *Proc. 2nd Int. Conf. Inventive Syst. Control (ICISC)*, (2018), 248–251. https://doi.org/10.1109/ICISC.2018.8399072
18. D. K. Chandra, V. Ravi, I. Bose, Failure prediction of dotcom companies using hybrid intelligent techniques, *Expert Syst. Appl.*, **36** (2009), 4830–4837. https://doi.org/10.1016/j.eswa.2008.05.047
19. F. Antunes, B. Ribeiro, F. Pereira, Probabilistic modeling and visualization for bankruptcy prediction, *Appl. Soft Comput.*, **60** (2017), 831–843. https://doi.org/10.1016/j.asoc.2017.06.043
20. H. Kim, H. Cho, D. Ryu, Corporate bankruptcy prediction using machine learning methodologies with a focus on sequential data, *Comput. Econ.*, **59** (2022), 1231–1249. https://doi.org/10.1007/s10614-021-10126-5
21. S. Smiti, M. Soui, Bankruptcy prediction using deep learning approach based on Borderline SMOTE, *Inf. Syst. Front.*, **22** (2020), 1067–1083. https://doi.org/10.1007/s10796-020-10031-6
22. S. Ben Jabeur, N. Stef, P. Carmona, Bankruptcy prediction using the XGBoost algorithm and variable importance feature engineering, *Comput. Econ.*, **61** (2023), 715–741. https://doi.org/10.1007/s10614-021-10227-1
23. A. Booth, E. Gerding, F. McGroarty, Automated trading with performance weighted random forests and seasonality, *Expert Syst. Appl.*, **41** (2014), 3651–3661. https://doi.org/10.1016/j.eswa.2013.12.009
24. D. Liang, C. C. Lu, C. F. Tsai, G. A. Shih, Financial ratios and corporate governance indicators in bankruptcy prediction: A comprehensive study, *Eur. J. Oper. Res.*, **252** (2016), 561–572. https://doi.org/10.1016/j.ejor.2016.01.012
25. G. Wang, J. Ma, S. Yang, An improved boosting based on feature selection for corporate bankruptcy prediction, *Expert Syst. Appl.*, **41** (2014), 2353–2361. https://doi.org/10.1016/j.eswa.2013.09.033
26. Y. C. Hu, A multivariate grey prediction model with grey relational analysis for bankruptcy prediction problems, *Soft Comput.*, **24** (2020), 4259–4268. https://doi.org/10.1007/s00500-019-04191-0
27. S. Y. Kim, A. Upneja, Predicting restaurant financial distress using decision tree and AdaBoosted decision tree models, *Econ. Model.*, **36** (2014), 354–362. https://doi.org/10.1016/j.econmod.2013.10.005
28. J. H. Min, Y. C. Lee, Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters, *Expert Syst. Appl.*, **28** (2005), 603–614. https://doi.org/10.1016/j.eswa.2004.12.008
29. J. Uthayakumar, N. Metawa, K. Shankar, S. K. Lakshmanaprabu, Financial crisis prediction model using ant colony optimization, *Int. J. Inf. Manage.*, **50** (2020), 538–556. https://doi.org/10.1016/j.ijinfomgt.2018.12.001

30. C. H. Wang, J. Z. Chen, Combining hidden Markov models with probabilistic Bayes networks to conduct business forecasting and risk simulation, *Soft Comput.*, **25** (2021), 8773–8784. https://doi.org/10.1007/s00500-021-05784-4
31. C. Lohmann, S. Möllenhoff, T. Ohliger, Nonlinear relationships in bankruptcy prediction and their effect on the profitability of bankruptcy prediction models, *J. Bus. Econ.*, **93** (2023), 1661–1690. https://doi.org/10.1007/s11573-022-01130-8
32. Q. Yu, Y. Miche, E. Séverin, A. Lendasse, Bankruptcy prediction using Extreme Learning Machine and financial expertise, *Neurocomputing*, **128** (2014), 296–302. https://doi.org/10.1016/j.neucom.2013.01.063
33. J. L. Bellovary, D. E. Giacomino, M. D. Akers, A review of bankruptcy prediction studies: 1930 to present, *J. Financ. Educ.*, **33** (2007), 1–42.
34. Y. Wu, C. Gaunt, S. Gray, A comparison of alternative bankruptcy prediction models, *J. Contemp. Account. Econ.*, **6** (2010), 34–45. https://doi.org/10.1016/j.jcae.2010.04.002
35. E. I. Altman, M. Balzano, A. Giannozzi, S. Srhoj, The Omega score: An improved tool for SME default predictions, *J. Int. Counc. Small Bus.*, **4** (2023), 362–373. https://doi.org/10.1080/26437015.2023.2186284
36. J. M. Liberti, M. A. Petersen, Information: Hard and soft, *Rev. Corp. Finance Stud.*, **8** (2019), 1–41. https://doi.org/10.1093/rcfs/cfy009
37. R. Gao, S. Cui, Y. Wang, W. Xu, Predicting financial distress in high-dimensional imbalanced datasets: A multi-heterogeneous self-paced ensemble learning framework, *Financ. Innov.*, **11** (2025), 50. https://doi.org/10.1186/s40854-024-00745-w
38. L. Zhu, Z. Zhang, M. J. C. Crabbe, Exploring small-scale optimization coupling learning approaches for enterprises' financial health forecasts, *Financ. Innov.*, **11** (2025), 78. https://doi.org/10.1186/s40854-024-00748-7
39. B. Branch, The costs of bankruptcy: A review, *Int. Rev. Financ. Anal.*, **11** (2002), 39–57. https://doi.org/10.1016/S1057-5219(01)00068-0
40. A. Bris, I. Welch, N. Zhu, The costs of bankruptcy: Chapter 7 liquidation versus Chapter 11 reorganization, *J. Finance*, **61** (2006), 1253–1303. https://doi.org/10.1111/j.1540-6261.2006.00872.x
41. F. Mai, S. Tian, C. Lee, L. Ma, Deep learning models for bankruptcy prediction using textual disclosures, *Eur. J. Oper. Res.*, **274** (2019), 743–758. https://doi.org/10.1016/j.ejor.2018.10.024
42. J. Gamboa, Deep learning for time-series analysis, *arXiv preprint*, arXiv:1701.01887, 2017.
43. A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architectures, *Neural Netw.*, **18** (2005), 602–610. https://doi.org/10.1016/j.neunet.2005.06.042
44. S. Siami-Namini, N. Tavakoli, A. Siami Namin, A comparison of ARIMA and LSTM in forecasting time series, *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, (2018), 1394–1401.
45. S. Lawrence, C. L. Giles, A. C. Tsoi, A. D. Back, Face recognition: A convolutional neural-network approach, *IEEE Trans. Neural Netw.*, **8** (1997), 98–113. https://doi.org/10.1109/72.554195

46. A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Commun. ACM*, **60** (2017), 84–90. https://doi.org/10.1145/3065386
47. S. Ghosh, D. L. Reilly, Credit card fraud detection with a neural-network, *Proc. Hawaii Int. Conf. Syst. Sci.*, **3** (1994), 621–630. https://doi.org/10.1109/HICSS.1994.323314
48. N. Kalchbrenner, E. Grefenstette, P. Blunsom, A convolutional neural network for modelling sentences, *arXiv preprint*, arXiv:1404.2188, 2014.
49. J. B. Yang, M. N. Nguyen, P. P. San, X. L. Li, S. Krishnaswamy, Deep convolutional neural networks on multichannel time series for human activity recognition, *Proc. IJCAI*, (2015), 3995–4001.
50. H. H. Nguyen, J. L. Viviani, S. Ben Jabeur, Bankruptcy prediction using machine learning and Shapley additive explanations, *Rev. Quant. Finance Account.*, **65** (2025), 107–148. https://doi.org/10.1007/s11156-023-01192-x
51. T. Hosaka, Bankruptcy prediction using imaged financial ratios and convolutional neural networks, *Expert Syst. Appl.*, **117** (2019), 287–299. https://doi.org/10.1016/j.eswa.2018.09.039
52. M. J. Kim, D. K. Kang, Ensemble with neural networks for bankruptcy prediction, *Expert Syst. Appl.*, **37** (2010), 3373–3379. https://doi.org/10.1016/j.eswa.2009.10.012
53. M. D. Odom, R. Sharda, A neural network model for bankruptcy prediction, *Proc. IJCNN Int. Joint Conf. Neural Netw.*, (1990), 163–168.
54. Z. R. Yang, M. B. Platt, H. D. Platt, Probabilistic neural networks in bankruptcy prediction, *J. Bus. Res.*, **44** (1999), 67–74. https://doi.org/10.1016/S0148-2963(97)00242-7
55. Y. Qu, P. Quan, M. Lei, Y. Shi, Review of bankruptcy prediction using machine learning and deep learning techniques, *Procedia Comput. Sci.*, **162** (2019), 895–899. https://doi.org/10.1016/j.procs.2019.12.065
56. I. Guyon, J. Weston, S. Barnhill, Gene selection for cancer classification using support vector machine, *Mach. Learn.*, **46** (2002), 389–422. https://doi.org/10.1023/A:1012487302797
57. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural Comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
58. D. N. T. How, C. K. Loo, K. S. M. Sahari, Behavior recognition for humanoid robots using long short-term memory, *Int. J. Adv. Robot. Syst.*, **13** (2016), 1–14. https://doi.org/10.1177/1729881416663369
59. H. Palangi, R. Ward, L. Deng, Distributed compressive sensing: A deep learning approach, *IEEE Trans. Signal Process.*, **64** (2016), 4504–4518. https://doi.org/10.1109/TSP.2016.2557301
60. W. Bao, J. Yue, Y. Rao, A deep learning framework for financial time series using stacked autoencoders and long-short term memory, *PLoS ONE*, **12** (2017), e0180944. https://doi.org/10.1371/journal.pone.0180944
61. M. Schuster, K. K. Paliwal, Bidirectional recurrent neural networks, *IEEE Trans. Signal Process.*, **45** (1997), 2673–2681. https://doi.org/10.1109/78.650093
62. T. Chen, R. Xu, Y. He, X. Wang, Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN, *Expert Syst. Appl.*, **72** (2017), 221–230. https://doi.org/10.1016/j.eswa.2016.10.065

63. A. Graves, N. Jaitly, A. Mohamed, Hybrid speech recognition with deep bidirectional LSTM, *Proc. IEEE Workshop Autom. Speech Recognit. Underst. (ASRU)*, (2013), 273–278. https://doi.org/10.1109/ASRU.2013.6707742
64. A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, J. Schmidhuber, A novel connectionist system for unconstrained handwriting recognition, *IEEE Trans. Pattern Anal. Mach. Intell.*, **31** (2009), 855–868. https://doi.org/10.1109/TPAMI.2008.137
65. Y. Fan, Y. Qian, F. L. Xie, F. K. Soong, TTS synthesis with bidirectional LSTM based recurrent neural networks, *Proc. Interspeech*, (2014). https://doi.org/10.21437/Interspeech.2014-443
66. Y. Lecun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel, Handwritten digit recognition with a back-propagation network, *Adv. Neural Inf. Process. Syst.*, **2** (1989), 396–404.
67. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, *Proc. IEEE*, **86** (1998), 2278–2323. https://doi.org/10.1109/5.726791
68. D. Fuqua, T. Razzaghi, A cost-sensitive convolution neural network learning for control chart pattern recognition, *Expert Syst. Appl.*, **150** (2020), 113275. https://doi.org/10.1016/j.eswa.2020.113275
69. A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet classification with deep convolutional neural networks, *Adv. Neural Inf. Process. Syst.*, **25** (2012).
70. Y. Zheng, Q. Liu, E. Chen, Y. Ge, J. L. Zhao, Time series classification using multi-channels deep convolutional neural networks, *Lect. Notes Comput. Sci.*, **8485** (2014), 298–310. https://doi.org/10.1007/978-3-319-08010-9_33
71. V. Agarwal, R. Taffler, Comparing the performance of market-based and accounting-based bankruptcy prediction models, *J. Banking Finance*, **32** (2008), 1541–1551. https://doi.org/10.1016/j.jbankfin.2007.07.014
72. T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, (2016), 785–794. https://doi.org/10.1145/2939672.2939785
73. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, et al., LightGBM: A highly efficient gradient boosting decision tree, *Adv. Neural Inf. Process. Syst.*, **30** (2017), 3146–3154.
74. F. Chollet, *Keras*, 2015.
75. M. Abadi, A. Agarwal, P. Barham, et al., TensorFlow: Large-scale machine learning on heterogeneous distributed systems, *arXiv preprint*, arXiv:1603.04467, 2015.
76. C. Elkan, The foundations of cost-sensitive learning, *Proc. 17th Int. Joint Conf. Artif. Intell. (IJCAI)*, (2001), 973–978.
77. F. Provost, Machine learning from imbalanced data sets 101, *Proc. AAAI'2000 Workshop Imbalanced Data Sets*, (2000).
78. V. S. Sheng, C. X. Ling, Thresholding for making classifiers cost-sensitive, *Proc. 21st Natl. Conf. Artif. Intell. (AAAI)*, (2006).
79. S. Nanda, P. Pendharkar, Linear models for minimizing misclassification costs in bankruptcy prediction, *Intell. Syst. Account. Finance Manage.*, **10** (2001), 155–168. https://doi.org/10.1002/isaf.203
80. S. M. Lundberg, S. I. Lee, A unified approach to interpreting model predictions, *Adv. Neural Inf. Process. Syst.*, 30 (2017).

81. F. Jiménez, J. Palma, G. Sánchez, D. Marín, M. D. Francisco Palacios, M. D. Lucía López, Feature selection based multivariate time series forecasting: An application to antibiotic resistance outbreaks prediction, *Artif. Intell. Med.*, **104** (2020), 101818. https://doi.org/10.1016/j.artmed.2020.101818
82. T. Niu, J. Wang, H. Lu, W. Yang, P. Du, Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting, *Expert Syst. Appl.*, **148** (2020), 113237. https://doi.org/10.1016/j.eswa.2020.113237

AIMS Press