Aleksander Kaspersen
Olav Lindemark

Interpretable Deep Learning for Bankruptcy Prediction

A Study of Norwegian SMEs Using LSTM Networks and the SHAP Framework

Master's thesis in Economics and Business Administration
Supervisor: Arild Brandrud Næss
June 2022

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Economics and Management
NTNU Business School

NTNU
Norwegian University of
Science and Technology

Aleksander Kaspersen
Olav Lindemark

Interpretable Deep Learning for Bankruptcy Prediction

A Study of Norwegian SMEs Using LSTM Networks and the SHAP Framework

Master's thesis in Economics and Business Administration
Supervisor: Arild Brandrud Næss
June 2022

Norwegian University of Science and Technology
Faculty of Economics and Management
NTNU Business School

NTNU
Norwegian University of
Science and Technology

Interpretable Deep Learning for Bankruptcy Prediction

A Study of Norwegian SMEs Using LSTM Networks
and the SHAP Framework

Olav Lindemark and Aleksander Kaspersen

June 2022

Abstract

The financial failure of a firm causes considerable losses to both the business community and society as a whole. Consequently, bankruptcy prediction has been a field of great interest and importance for academics and practitioners alike. In recent years, more opaque machine and deep learning methods have been developed, proven to have superior predictive performance compared to simpler machine learning models. Still, simpler bankruptcy prediction models are often preferred for use in practice due to the black box problem, encompassing reduced interpretability and trustworthiness.

In this thesis we explore the use of long short-term memory (LSTM) networks capable of utilizing sequential accounting data for bankruptcy prediction, while focusing on interpretability using the Shapley Additive Explanations (SHAP) framework. Further, to evaluate the predictive performance of the LSTM networks, we create a recurrent neural network (RNN) and a fully connected feed-forward neural network. The networks are trained on a real-world dataset of Norwegian small and medium sized enterprises (SME). The dataset consists of 212 020 unconsolidated financial statements from 2006–2019, used to construct 156 predictor variables. Only a small percentage of the financial statements (0.5665%) were classified as bankrupt in the period, meaning the dataset is severely imbalanced. To account for this, we implement a cost-sensitive learning strategy in the training of all the deep neural networks.

The LSTM network using a sequence of four accounting years and all features, obtained an AUC and a Brier score of 0.9288 and 0.0477, respectively. This was an increase of 5.56% in AUC and decrease of 65.36% in Brier score compared to the fully connected feed-forward neural network. Moreover, the LSTM network using a subset of 30 features and a sequence of four years achieved an increase of 1.74% in AUC compared to the RNN. This indicates that LSTM networks have higher predictive performance than the baseline neural networks. We further observed a decrease in predictive performance for each time step omitted from the LSTM networks, indicating that longer sequences of data better enables the LSTM networks to predict bankruptcy. To enhance model interpretability, we implement SHAP to explain individual predictions and to give insight into the general logic of the model. Moreover, we evaluate whether the learned behavior of the LSTM networks is consistent with economic theory and discuss the framework's capabilities from a financial institution and decision-making perspective. Our findings suggest that SHAP increases the interpretability of deep neural networks, and therefore facilitates adoption of high-performing LSTM networks for bankruptcy prediction in the financial services sector.

Sammendrag

Konkurs fører til betydelige tap for både næringsliv og samfunnet som helhet. Derfor har konkursprediksjon vært et viktig tema for akademikere, finansielle institusjoner, bedriftsledelse og andre interessenter. I de siste årene, har flere maskin- og dyplæringsmetoder med gode prediktive evner blitt utviklet. Likevel, på grunn av black–box–problemet, som medfører redusert tolkbarhet og pålitelighet, blir gjerne enklere maskinlæringsmodeller foretrukket for bruk i den virkelige verden.

I denne oppgaven utforsker vi bruken av “Lang korttidsminne” (LSTM) nettverk, i stand til å bruke sekvensiell regnskapsdata for konkursprediksjon. For å forbedre tolkbarheten av LSTM nettverkene implementerte vi Shapley Additive Explanations-rammeverket (SHAP). Videre, for å evaluere de prediktive evnene til LSTM nettverkene, konstruer vi et rekurrent nevralt nettverk (RNN), og et flerlags forovermatet nevralt nettverk. Modellene blir trent på et datasett med små og mellomstore norske bedrifter (SMB). Datasettet består av 212 020 ukonsoliderte årsregnskap fra perioden 2006–2019, som vi brukte til å konstruere 156 variabler. Kun en liten prosentandel av regnskapene (0,5665%) var klassifisert som konkurs i perioden, noe som betyr at datasettet er svært ubalansert. For å ta høyde for dette, ble en kostnadssensitiv læringsstrategi brukt i treningen av de nevrale nettverkene.

LSTM-nettverket som brukte en sekvens på fire regnskapsår og alle variabler oppnådde en AUC og Brier score på respektive 0.9288 og 0.0477, en økning på 5,56% i AUC og reduksjon på 65,36% sammenlignet med det forovermatede nevrale nettverket. I tillegg oppnådde LSTM-nettverket som brukte en sekvens på fire regnskapsår og 30 variabler en økning på 1,74% i AUC sammenlignet med RNN. Dette indikerer at LSTM-nettverkene har høyere prediktive evner enn de andre nevrale nettverkene. Videre observerte vi en reduksjon av prediktive evner når vi reduserte sekvenslengden, noe som indikerer at lengre sekvenser av data øker LSTM-nettverkene sin evne til å predikere konkurs. For å øke tolkbarheten av LSTM-nettverkene, brukte vi SHAP-rammeverket for å forklare individuelle prediksjoner, samt for å gi innsikt til den generelle oppførselen til modellene. Videre sammenlignet vi om den lærte oppførselen til LSTM-nettverkene var i tråd med økonomisk teori. I tillegg diskuterte vi SHAP-rammeverkets evner fra et finansinstitusjons- og beslutningstakingsperspektiv. Funnene våre tyder på at SHAP-rammeverket øker tolkbarheten av dype nevrale nettverk, og derfor kan fasilitere bruk av komplekse, høytytende LSTM-nettverk for konkursprediksjon i næringslivet.

Preface

This thesis is written as part of our Master of Science degree in Economics and Business administration, concluding our five year journey at Norwegian University of Technology (NTNU). We will cherish our time and the relationships we developed here for the rest of our lives.

First and foremost, we want to thank our supervisor Arild Brandrud Næss for invaluable insight and guidance throughout the whole process. Secondly, our gratitude goes to Ranik Raaen Wahlstrøm for providing the data utilized in this thesis, and for being an all round valuable resource.

We would also like to thank our families and friends for their support throughout our studies. Special thanks also goes to Maja and Solveig whom we had the pleasure of sharing an office with for the past months.

We take full responsibility for the content of this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Corporate bankruptcy leads to considerable losses to both the business community and society as a whole (Van Gestel et al., 2003). Therefore, bankruptcy prediction is a field of great interest and importance to researchers, financial institutions, company decision-makers, employees, investors and other stakeholders (Shi & Li, 2019b; Smiti & Soui, 2020; Tang et al., 2019; Van Gestel et al., 2003). Consequently, this has resulted in the development of several distinct financial failure forecasting tools providing company management the ability to make timely strategic decisions to prevent further financial distress. Likewise, financial institutions such as banks can leverage these tools to reduce their cost of capital by efficiently and automatically predicting their clients' default risk (Härdle et al., 2009; Van Gestel et al., 2003).

Due to the availability of more data and increased computing power, we have in the past decade been witnessing the development of more complex machine learning methods such as deep neural networks. Despite some of these state-of-the-art models showing superior performance compared to more traditional statistical models, are they often criticized for being black box methods, meaning they allow limited transparency into the decision process. This in turn have restricted their use in disciplines such as finance and healthcare. Consequently, financial institutions still mostly use more simplistic models to predict bankruptcy (Zhang & Thomas, 2015) even though small performance increases can lead to significant economic gains (Stein, 2005).

Furthermore, the General Data Protection Regulation (GDPR) was adopted by the European Parliament and became law as of May 2018. One part of GDPR is the article concerning automated decision-making, which to some extent, introduces a right of explanation. This gives individuals a right to obtain "meaningful explanations of the logic involved" when automated decision making takes place (Art. 22 GDPR, 2016). Although the right of explanation in GDPR only concerns private individuals and not companies, it illustrates how interpretability in AI is a topic of growing concern, and how it rationalizes the use of more simplistic interpretable models.

However, simplistic models are not able to treat bankruptcy as a process, but only as a steady state, which induce some serious drawbacks. The failure of a company is not a sudden or unexpected event (Laitinen, 2005), but rather a result of several periods of adverse performance. Therefore, Kim et al. (2020), states multi-period sequential models are more appropriate for bankruptcy prediction.

Still, in order to utilize complex multi-period models in the real world, the lack of interpretability needs to be addressed. The research domain of explainable AI aims to increase interpretability and explainability of complex machine learning models while keeping high levels of predictive performance. A major contributor to such research is Lundberg and Lee (2017), who introduced the Shapley Additive explanations (SHAP) framework. The framework enables both global and local explanations, encompassing interpretations of feature effect and magnitude on model prediction based on Shapley values from coalition game theory. However, the use of this framework in the domain of deep learning neural networks for bankruptcy prediction is still scarce.

Despite the majority of global companies being small or medium-sized enterprises (SMEs), the literature on corporate failure prediction is mostly focused on large or listed companies (Gordini, 2014; Paraschiv et al., 2021). According to the European Commission (2021), 99.8% of all EU enterprises were categorized as SMEs in 2020. Combined, the SMEs were accountable for over 50% of the total value added produced by the EU. The lack of research on bankruptcy prediction amongst SMEs is mainly due to better availability of financial data for larger enterprises. Additionally, market-based information for privately held SMEs are often unavailable (Filipe et al., 2016). It would be mutually beneficial to both financial institutions and the SMEs themselves to improve their bankruptcy prediction models. Financial institution would reduce their risk exposure, resulting in improved financing options for SMEs through lower risk premiums (Tobback et al., 2017).

Moreover, data scarcity is a common problem for bankruptcy prediction; most recent studies employ data samples from only 400 companies or less (Veganzones & Severin, 2020). In many cases the available data is also limited in terms of quality due to the data often being sourced from, for example, a single set of clients from a specific bank, which may induce bias and distort the result. Additionally, bankruptcy is considered a rare event, meaning bankruptcy data usually has problems with imbalanced class distributions. Therefore, training machine learning models for bankruptcy prediction often requires implementations of data balancing strategies such as resampling methods. However, as Zmijewski (1984) demonstrates, the use of these strategies may also result in biased probability estimates, meaning model training on unprocessed data is to be preferred for real world applications.

1.1 Research questions

The motivation for the topic of this thesis is an accumulation of all previously described challenges in regards to the use of complex deep neural networks for bankruptcy prediction in the real world. The objective of this master thesis is therefore to discuss and utilize the SHAP framework to increase model interpretability and enhance real-world applications of deep recurrent neural networks for bankruptcy prediction. To achieve this, we train LSTM networks on sequential imbalanced data of all unconsolidated annual financial statements from Norwegian companies, in the accounting years 2006–2019. We further compare the predictive performance of the LSTM networks with a traditional RNN and a fully connected feed-forward network. This thesis will therefore have the following research questions:

- *To what extent can LSTM networks using sequential accounting data produce superior predictive performance compared to other neural network models for bankruptcy prediction?*

- *How can the SHAP framework increase interpretability of deep recurrent neural networks for bankruptcy prediction, and to what extent can this facilitate the adoption of deep learning for bankruptcy prediction in the financial services sector?*

1.2 Thesis structure

In chapter 2 all considerations and relevant background for deep neural networks for bankruptcy prediction are presented, including a description of the black box problem and explainable AI. Moreover, we present problems related to imbalanced datasets and the neglect of the time dimension in bankruptcy prediction. Lastly, relevant previous work is presented.

Chapter 3 describes the foundation of the models utilized in this thesis. Firstly, the basics of neural networks are presented, followed by a description of deep recurrent neural networks and LSTM networks. Secondly, a description of Shapley values and the SHAP framework for model interpretability is presented. Relevant evaluation metrics for bankruptcy prediction models are also introduced. Lastly, a cost-sensitive learning strategy to address the data imbalance is described.

Chapter 4 gives details of our specific choices in regards to methodology, based on the two previous chapters. We start by describing the data and data preprocessing, followed by a description of the training and test splitting scheme. We further outline the implementation of the LSTM neural networks, describing the chosen architecture and parameters. Moreover, we describe the implementation of the baseline neural networks. Lastly, we present the model evaluation metrics.

Further, in chapter 5, the results based on the evaluation metrics are presented. This is followed by results regarding the SHAP framework utilized to interpret the learned behaviour of the LSTM networks, including the feature impact magnitude and effect. Moreover, we present three individual predictions made by the LSTM networks.

The discussion regarding model performance is found in Chapter 6, followed by interpretations of the learned behaviour of the LSTM networks based on the SHAP framework. Moreover, we evaluate the SHAP frameworks ability to interpret specific predictions. Furthermore, we discuss the limitations of the thesis, ending with a discussion regarding the real-world implications of our analysis.

We outline the findings of the thesis in Chapter 7, before ultimately pointing out new directions for future work in the field of deep learning for bankruptcy prediction, especially in regards to real-world applications.

1.3 Contributions

This thesis has evaluated the use of the SHAP framework for increased interpretability of deep LSTM networks for bankruptcy prediction in order to increase applicability of complex deep neural networks in the financial services sector.

Though we are not the first to utilize deep LSTM network for bankruptcy prediction, the research within the domain is lacking. Therefore, by demonstrating the predictive performance of a deep LSTM networks, we further confirm the usefulness of complex machine learning models capable of utilizing sequential data within the field.

To the best of our knowledge, this thesis represents the first attempt to utilize cost-sensitive learning as a solution to the imbalanced data problem for deep recurrent neural networks for bankruptcy prediction as motivated by Zmijewski (1984). We found the strategy to be a feasible alternative to resampling methods, with the benefit of not altering data.

Still, the main contribution of this thesis comes from the testing and discussion of the SHAP framework to enhance interpretability of deep LSTM networks. We found SHAP to be a viable tool for reducing the black box problem, facilitating adoption of LSTM networks in the financial services sector.

Chapter 2

Theoretical background and previous work

In this chapter the theoretical background in regards to bankruptcy prediction, deep learning, the black box problem and the characteristics of bankruptcy data is presented. We also present previous work regarding bankruptcy prediction, feature effects, interpretability, data balancing and time sensitive modeling.

2.1 Bankruptcy prediction

Bankruptcy prediction models have been a staple credit risk management tool for both investors and creditors alike over the past few decades (Härdle et al., 2009). Throughout time, such financial forecasting tools have been gradually developed and improved. A notable event that had considerable impact on the development of bankruptcy prediction models was the financial crisis of 2008–2009, where numerous companies either went bankrupt or experienced financial distress. This became a wake up call for regulators, practitioners, and other company stakeholders, accelerating research in the domain of bankruptcy prediction (Shi & Li, 2019a). Another factor that contributed to the development of bankruptcy prediction models, is the Basel II framework ("Basel II", 2004), which established new risk and capital management requirements for the banking sector. Because the minimum capital requirements for companies are calculated using existing bankruptcy prediction models, banks are incentivized to refine their models in order to lower their capital requirements in the future (Kirkos, 2015). This interest and research has resulted in a number of different bankruptcy prediction models being developed over the years (Gupta & Chaudhry, 2019).

2.1.1 Definition of bankruptcy

Bankruptcy for companies in Norway is regulated through The Bankruptcy Act of 1984. §61 of this act states that "The debtor is insolvent when he/she cannot meet his/her obligations as they fall due unless this insolvency may be assumed to be of a transient nature". Furthermore, §60 states that "If the debtor is insolvent, the person in question's estate shall be subject to bankruptcy proceedings when the debtor or a creditor so requests" (The Bankruptcy Act, 60–61, 1984). In this thesis, bankruptcy is considered a binary event, determined by our dichotomous target variable $\texttt{bankrupt}_{\text{fs}}$. This variable is described in section 4.1.1.

2.1.2 Machine learning for bankruptcy prediction

In classical programming, humans construct and define rules which an algorithm uses to analyze data and produce answers (output). On the other hand, in a machine learning (ML) system, the human inputs the data and the answers, and the model outputs the rules. This means that the algorithm (or system) is trained rather than explicitly programmed (Chollet, 2018). These rules (models) can in turn be applied to new data, with the main areas of focus being prediction, classification and clustering (Athey, 2018).

The use of machine learning algorithms is closely related to Data-driven decision making, suggested by Brynjolfsson et al. (2011) to increase output and profitability of businesses. Artificial intelligence (AI) and machine learning systems have therefore emerged a staple of operations within multiple sectors utilizing new information technologies for increased business value (Barredo Arrieta et al., 2020). Further, for company stakeholders and financial institutions, financial forecasting tools such as bankruptcy prediction models have become a valuable asset.

Still, to understand why machine learning models have become so important, we need to understand what they do. A machine learning model transforms input data into outputs. How to transform the data (rules) is learned by the model through continuous exposure to input data with known outputs. Thus, the core problem for a machine learning model is learning to transform the input data into more easily understandable and usable representations of the data. These *representations* are simply another way for the model to look at the data, that enables it to better understand what the information indicate and thereby produce the correct output (Chollet, 2018). This way of learning are referred to as representation learning (Bengio et al., 2014).

For a machine learning model for bankruptcy prediction to be considered effective, two criteria need to be satisfied. First of all, the model must be accurate in its predictions. Secondly, the model needs to be interpretable (Son et al., 2019). However, there is a trade-off between predictive performance and model interpretability (Došilović et al., 2018), and how to prioritize these criteria depends on the situation. As previously mentioned, financial forecasting tools provide companies the ability to make decisions to prevent further financial distress, while financial institutions such as banks can leverage the predictions to determine their clients' default risk. These different applications of bankruptcy prediction models do not necessarily have the same prioritization between accuracy and interpretability. To elaborate, lets give an example. When a bank is deciding whether to issue a loan, the primary criterion is prediction accuracy, as the banks main objective is to reduce their own risk. To do this, they do not necessarily need to know why the model came to the decision it did. Meanwhile, a company's leadership is interested in the reasoning behind the prediction. This in order to know what decisions need to be made to reduce the likelihood of further financial distress. Hence, for such a case, the tool not only needs to be accurate, but also interpretable (Alaka et al., 2018). A more detailed description of issues related interpretability can be found in Section 2.2.

2.1.3 Deep learning for bankruptcy prediction

Deep learning is a subfield of machine learning that similarly aims to learn how to create the most appropriate representations of the data, that best enables the model to do the task at hand. However, a deep learning model focuses on learning "successive layers of increasingly meaningful representations" (Chollet, 2018) in order to learn high-level nuances and patterns not necessarily recognized by traditional machine learning methods. This means that a deep learning model aims to produce multiple layers of representations (Guo et al., 2016). More often than not, these layers of representations are learned via neural networks. Neural networks are stacked layers of interconnected computational nodes that through training learn to transform and combine input data into meaningful outputs (representations) to be further transmitted through the network. This way the model transform representations at one level to representations of a higher level (LeCun et al., 2015), being the main reason why deeper networks tend to be better at learning representations in data than traditional machine learning models (Razavi, 2021). The amount of stacked layers are the depth of the model, where modern deep models can have hundreds of successive layers (Chollet, 2018).

There are multiple different types of neural networks. The most basic are fully connected feed-forward neural networks, also referred to as densely connected neural networks (Chollet, 2018). "Fully connected" refers to the fact that each node in a layer are connected to all nodes in the next layer. Moreover, "feed-forward" imply that there are no backwards connections between the layers (Gawehn et al., 2016). A more in-depth description of a fully connected network is presented in Section 3.1.1.

Another type of neural networks is convolutional neural networks (CNNs). These networks specializes in image recognition (O'Shea & Nash, 2015), and are designed to process data in forms of multiple arrays (LeCun et al., 2015).

Recurrent neural networks (RNNs) specializes in processing sequential data, and are often used for language and speech recognition. RNNs process the elements of the sequence one at a time, while calculating a *hidden state* that contains information about the previous elements of the sequence (Chollet, 2018; LeCun et al., 2015). However, traditional RNNs often struggles to retain long-term dependencies in data. Therefore, Hochreiter and Schmidhuber (1997) created a special type of RNN called *long short-term memory* (LSTM) networks that in addition to the hidden state introduces a *cell state* that enhances the models capabilities of "remembering" information for longer periods of time. A more thorough description of RNNs and LSTM networks are presented in Sections 3.1.6 and 3.1.7. In this thesis, we focus on using LSTM networks for bankruptcy prediction in order to process accounting data over multiple periods (sequential accounting data).

Neural networks, and specifically deep learning, have become an important topic in both academic research and practical applications spanning multiple fields (Qu et al., 2019). Still, Qu et al. (2019) state that research on the application of deep learning in finance and management is lacking. An overview of the literature is found in Section 2.5.1.

2.1.4 Accounting-based predictor variables

The causes of bankruptcies and financial distress are numerous, including: macroeconomic and industry specific factors, governance and managerial problems, political events and laws, and even pandemics and wars. Even though there are many causes to bankruptcy and these causes often are easily determined, capturing such information in terms of data is difficult because each cause cannot be entirely reduced to a single, measurable parameter. Moreover, information regarding the previously listed causes are often not available for every company, making such data difficult to use for modeling.

Despite the fact that corporate failure prediction has been a subject of extensive research, the literature is largely inconclusive and contradictory on what types of features are the key determinants of bankruptcy. There is strong evidence that the best predictor variables differ significantly between data samples (Balcaen & Ooghe, 2006) and between countries in terms of their interactions and influence (Filipe et al., 2016). Because of this, finding predictive variables that apply for all populations of firms is difficult, and prediction models applied outside the original context may not be as accurate. To obtain accurate information regarding the population in question, the modeling therefore needs to be data specific.

In order to address the issue of determining what types of features to use for bankruptcy prediction, accounting based financial ratios are most commonly used, because they often provide a relatively objective, quantitative measure of a company's financial situation and performance (Balcaen & Ooghe, 2006; Veganzones & Severin, 2020). A financial ratio is simply a ratio where both the numerator and the denominator are accounting items retrieved from the financial statements of a firm, and therefore provide information regarding the financial situation of a company (Nadar & Wadhwa, 2019). Accounting data are both easily available through annually public financial statements of the companies and they are reliable. One of the main strengths of financial ratios compared to raw entries springs form their ability to control for companies size effect (Barnes, 1987; Salmi & Martikainen, 1994).

Other advantages of using financial ratios to predict bankruptcy are their ability to control for industry-wide factors (Barnes, 1987). Additionally, bankruptcies are often the result of several years of adverse performance and will therefore largely be captured by the firm's accounting statements, whereas the relationship between corporate failure and alternative predictor variables such as corporate governance measurements are more ambiguous and challenging to identify. Loan covenants are generally based on accounting numbers and this information is more likely to be reflected in accounting-ratio-based models (Agrawal et al., 2018). Furthermore, the International Financial Reporting Standards (IFRS) promote the comparability of financial statements internationally, thus aiding the development of more widely generalizable bankruptcy prediction models between countries.

2.2 The black box problem

Significant concerns about the moral hazards associated with the increasing prevalence of algorithms and machine learning as a substitute for human judgement in decisions within financial services and consumer credit ratings has been raised in recent years. The concern stems from the opacity into the inner workings of deep machine learning models, meaning there is an absence of mechanisms to reproduce or explain the decision-making processes of a given model (von Eschenbach, 2021). Consequently, understanding the reason a machine learning model reaches a decision becomes difficult when the "ex ante predictions and ex post assessment of the system's operations is difficult to formulate precisely" (Zerilli et al., 2019). This is the essence of the black box problem.

A machine learning algorithm can generally be opaque in two different ways: (1) the process or mechanism for how machine learning arrives at outputs from given inputs may be inaccessible or unknowable, and (2) inputs themselves may be unknown to programmer or observers (von Eschenbach, 2021). These types of opacity can stem from intentional company secrecy, technical illiteracy or the characteristics of the algorithm (Burrell, 2016).

2.2.1 Explainable AI

Considering that the prediction accuracy of machine learning can be superior to human judgement, its use is not in itself problematic. The troublesome part is when opaque machine learning models is used for decisions that substantially influences people's lives, and the result or decision made by the model are difficult, if not impossible, to dispute or appeal. Additionally, those who suffer the consequences of the decisions often lack recourse to address them (von Eschenbach, 2021). Moreover, the black box problem for Deep Learning algorithms is enhanced when the outcomes are ethically problematic, and are based on biased algorithms or decision models. Especially when these biases can not be detected and therefore not be addressed nor accounted for.

Explainable AI (XAI) aims to produce ML models and techniques to address this problem without reducing the predictive effectiveness (Barredo Arrieta et al., 2020). The goals are therefore to create ML models that (1) are explainable while maintaining a high level of prediction accuracy, and (2) enables humans to trust and understand the emerging generation of ML and AI (Barredo Arrieta et al., 2020). In part, this is related to increasing model interpretability.

2.2.2 Interpretability

Interpretability is the degree to which a human can understand the cause of a decision (Biran & Cotton, 2017). This is important as in general, humans are hesitant to utilize techniques that are not directly interpretable, tractable and trustworthy (Barredo Arrieta et al., 2020). For instance, Oxborough et al. (2018) found that 67% of business leaders believes that AI and automation will impact negatively on stakeholders trust levels in their industry in the next five years. There is also a trade-off between predictive performance and transparency in a model (Došilović et al., 2018), meaning that a sole focus on performance will increase the opaqueness of the systems. Though performance is important for bankruptcy predication as small performances increases can lead to great increases in profitability (Stein, 2005), an improvement in the understanding of a system can lead to corrections of its deficiencies. Therefore, taking interpretability into consideration in the development of ML models, can improve its applicability and reduce the ethical challenges described in Section 2.2.1, mainly for 3 reasons (Barredo Arrieta et al., 2020):

- Interpretability helps ensure impartiality in decision making. This encompasses detecting, and correcting from bias in the training of the model
- Interpretability helps in highlighting potential adverse perturbations, and consequently robustness.
- Interpretability can act as insurance that only relevant and significant variables are used to produce the output. This means that interpretability helps ensure casuality in the model reasoning.

Methods for machine learning interpretability are often grouped into two categories, namely: intrinsic interpretability and post-hoc interpretability, depending on the time when the interpretability is obtained. *Intrinsic interpretability* refers to machine learning models that are self-explanatory, meaning they are interpretable due to their simple structure. Some of these models include simple decision trees, rule-based models and linear models (Molnar, 2022, Chapter 3.2). *Post-hoc interpretability* on the other hand, refers to the application of interpretation methods after model training. This therefore requires creating a second model to provide an explanation of the existing model (Du et al., 2019). Due to the lack of intrinsic interpetability of most deep learning methods, post-hoc interpretability methods are needed. In general, there are two types post-hoc explanations: global and local explanation models.

The idea behind *post-hoc global explanation* is that a machine learning model through training automatically learn useful patterns, and store this knowledge in the structure and parameters of the model (Du et al., 2019). The goal of a post-hoc global explanation model is therefore to access this knowledge, and reveal the learned model behaviour. Hence, global explanation models give insight into the general logic used by the model for making predictions (Demajo et al., 2020). Managers and decision makers are interested in using bankruptcy prediction models as management tools. Therefore, they require the model to give insight into the general logic of the model, meaning general importance of features, and their effect on model predictions. In such a manner, the model can be used to give valuable insight into their financial situation, and work as a guideline for which aspect of their business needs improvement. This in turn ensures better allocation of company resources. *Post-hoc local explanation* models focus on identifying the impact of each specific input feature on a specific model prediction. Customers and companies applying for loans are mostly interested in the reason behind why their loan application was denied, meaning they want to know the model reasoning of a specific prediction (Demajo et al., 2020). Moreover, company managers and decision makers can leverage local explanations to evaluate their financial situation and consequently make more informed decisions. Therefore, local explanations are preferred in this context.

As mentioned in Section 2.1.2, for financial institutions such as banks, will their main concern be model accuracy, rather than interpretability. A reason for this is that they are the ones taking on most of the risk when it comes to issuing credit. Misclassifying a bankrupt company as healthy (type I error) is the most costly for investors and creditors as the debt will not be reimbursed, while classifying a healthy company as bankrupt (type II error) can lead to lost earnings in terms of interest (du Jardin, 2015; Lohmann & Ohliger, 2019; Stein, 2005; Trinkle & Baldwin, 2007). While a model cannot eliminate both errors, a small percentage improvement in accuracy can materially impact the lending institution's profit (Trinkle & Baldwin, 2007). However, as previously stated, improvements of interpretability may lead to corrections of model deficiencies. Hence, increased interpretability may also improve model performance and trustworthiness, consequently facilitating adoption for financial institutions. Additionally, financial institutions may desire to give their customers and clients explainable reasons for loan denial, increasing customer trust and loyalty.

For the company stakeholders, all these considerations can be narrowed down to three questions, formulated by Bracke et al. (2019).

- What drove the explanations more generally?
- Which features with what effect mattered in individual predictions?
- How does the model work, and can it be easily explained?

When evaluating the interpretability of deep neural networks for bankruptcy prediction, these questions should be addressed. Note that these questions were originally created for default risk prediction, but are still highly relevant for bankruptcy prediction models. Additionally, these questions have been slightly modified for our use.

2.2.3 Interpretability methods

To increase the interpretability of black box models, several method have been proposed, This includes LIME, DeepLIFT, Layer-Wise relevance propagation, classic Shapley value estimation, and SHAP. In this section we will briefly introduce two of the most popular ones, namely the Local interpretable model-agnostic explanations (LIME) and the Shapley additive explanations (SHAP) of Lundberg and Lee (2017).

Local interpretable model- agnostic explanations

LIME is a post-hoc local explanation method aiming to provide explanations of any machine learning model. As the name suggest, the method focuses on local interpretations used to explain individual predictions (Molnar, 2022, Chapter 3.2). Instead of trying to understand the entire model at once, the method tweak the inputs of specific instances to see by how much the prediction changes. If the change is minuscule, the variable may not be an important predictor for that particular instance. Oppositely, if the difference is significant, the variable is of importance for the prediction. LIME has been proven to offer good approximations of the predictions locally. However, these approximations do not necessarily apply globally.

Shapley additive explanations

Shapley additive explanations (SHAP) was presented by Lundberg and Lee (2017) and is a unified (from six different models) framework for interpreting predictions of complex models. It came into fruition to address the problem of knowing how the distinct explanation models are related and when the different methods are preferred over another. SHAP quantifies the contribution each feature have on the prediction by the means of Shapley values from coalition game theory, providing a strong theoretical foundation to the framework (Molnar, 2022, Chapter 9.6). Unlike the LIME-framework, SHAP is also fit to provide global explanations for a model. In this thesis we will make use of the SHAP framework for model interpretations. A more detailed description of the SHAP and its capabilities is provided in Section 3.2.2.

2.3 The imbalanced dataset problem

Rare events are difficult to detect due to their infrequency. Nevertheless, misclassification of rare events can be costly (Haixiang et al., 2017). In the case of bankruptcy prediction, misclassification may lead to capital loss and even contagion effects. Further, the consequences may not only be individual, but cause a downward spiral for the whole economy, impacting related firms, employment and economic welfare (Veganzones & Séverin, 2018).

Veganzones and Séverin (2018) states that when a model is trained on data with a class imbalance ratio of 4:1 or higher, the models capability of predicting bankruptcy is at risk. In real world bankruptcy prediction this ratio of non-bankrupt to bankrupt companies can be as low as 100:1 or even 1000:1 (Veganzones & Séverin, 2018; Zhou, 2013). In the dataset utilized in this thesis, the proportion of bankrupt companies is 0.5665%, meaning it is highly imbalanced. Therefore, a major concern regarding our data characteristics is the imbalanced class distribution.

The main reasons why data imbalance causes decreased machine learning model performance, concerns the loss function of the classification algorithm (Kim et al., 2015). A widely used loss function named binary cross-entropy concerns arithmetic accuracy. This is the ratio of the number of correctly classified instances over the number of total instances. In other words, the objective of the models becomes to maximize the classification rate. Consequently, in the presence of greatly imbalanced data, the classifier tends to learn how to predict the majority class, rather than the minority class because the cost of making errors favour the majority class. To elaborate, lets give an example. If the class imbalance ratio is 100:1 and the model changes the decision boundary to correctly classify one more observation of the minority class, chances are that at least two or more majority class observations gets misclassified. Given an arithmetic loss function, this will result in a higher cost for the model than the other option of correctly classifying the two majority class observations while continuing to misclassify the minority. Consequently, if the model has to choose between the two options, it will favour the latter as it comes with a lower cost. In other words, the cost of making errors favour the majority class because the minority class has a lower prior probability and consequently a lower error cost. Even though a classification model trained like this can acquire higher prediction accuracy's than those also trying to consider the minority class, this seemingly good performance can be argued as being meaningless when the true error cost of the minority class is higher than it should be based on the data distribution, often being the case for bankruptcy prediction (Wang & Japkowicz, 2010). Because of this, arithmetic accuracy loss functions and metrics can be considered unfit for imbalanced datasets.

In the case of bankruptcy prediction, the problem of finding the correct decision boundary can be enhanced by two factors. Most models utilizes financial data as independent variables, as they give information about a firms financial situation, and have proven important considerations for the classification. Still, financial data can be manipulated, which may lead to distorted data where failing firms (minority) may invade the boundary of the non-failing firms (majority) reducing model performance (Veganzones & Séverin, 2018). Furthermore, firms with seemingly similar financial situations may not have the same fate, as companies may delay or even avoid bankruptcy if their environment is growing enough to support a resource-deficient firm, and proper managerial actions are taken (D'Aveni, 1989).

Several methods to solve this class imbalance issue have been proposed in the literature (Haixiang et al., 2017). They can generally be categorized into two strategic approaches: preprocessing and cost-sensitive learning. Preprocessing includes resampling, feature selection and feature extraction methods, of which there are several. Resampling methods include undersampling, oversampling and hybridsampling. These methods work by artificially altering the data to balance the dataset. Undersampling methods use all of the minority instances, and extract samples of the majority instances to balance the data. Oversampling is the opposite, and means increasing the number minority class instances to that (or close to that) of the majority class. This can be done by either duplicating minority instances, or artificially create new ones that mimic the original observations. Hybridsampling is a combination of the two methods, both increasing the number of minority instances and reducing the number of majority observations. In the case of financial distress estimation and prediction, Zmijewski (1984) illustrates the problems related to training bankruptcy prediction models on artificially balanced or changed data. He shows that changing the data introduces significant bias into the model. Therefore, resampling methods have significant drawbacks. Feature selection is often separated as another issue for imbalanced learning. However, in imbalanced scenarios there is a risk that the minority class is discarded as noise by the model. Removing irrelevant features has been shown to reduce the risk of treating minority samples as noise, and can therefore also be utilized for dealing with imbalanced data (Yijing et al., 2016).

The second strategy for dealing with class imbalance is cost-sensitive learning. This strategy works by assuming a higher cost of misclassification for the minority class than the majority class, and thereby making the model pay more attention to correctly classifying the minority class. This cost is usually specified in a cost matrix (Haixiang et al., 2017). Still, this way of dealing with imbalance is much less popular than resampling methods. The main reason, and this methods premier drawback, is the difficulty of setting the actual values of the cost matrix (Krawczyk et al., 2014). In most cases, the true cost of misclassification is not known from the data, and can not be given by an expert. Be that as it may, some strategies to deal with this issue have been proposed. López et al. (2015) suggests setting the majority class weight to 1, and the minority class penalty cost equal to the imbalanced ratio. Another similar method is suggested on the Tensorflow web page ("Classification on imbalanced data", 2022), and is presented in Section 3.4. Though cost-sensitive learning strategies do not necessarily alter the data, it still introduces bias into the model by assuming values for the cost matrix.

As the dataset utilized in this thesis is severely imbalanced, we need to utilize a strategy to deal with the imbalanced data problem. One of the objectives of this thesis is to train on realistic bankruptcy data for increased practical application, meaning resampling methods that alter the data is unfit for our purpose. Motivated by the statements of Zmijewski (1984) we therefore in this thesis focus on a cost-sensitive learning method presented in Section 3.4.

2.4 Neglect of the time dimension in bankruptcy prediction

Classical statistical prediction models often ignore the fact that companies change over time by only using one single observation (one annual account) in the estimation sample. This can both cause problems and limitations when predicting bankruptcy. The main assumption that consecutive annual account are independent, repeated measurements, are not met when only using one single observation in the prediction model. In fact, the observations are not entirely independent and it may be worthwhile to model this relationship (Balcaen & Ooghe, 2006).

The initial problem in this context relates to the fact that looking at one annual account, and the choice of when to observe it, may introduce bias to the model (Mensah, 1984; Shumway, 2001). For example, it may be possible for a model to classify a relatively healthy business that is suffering from a temporarily adverse situation as bankrupt.

Furthermore, a model that only use observations one year prior to failure should be restricted to only predict bankruptcy one year in the future ($t + 1$), because such a model is likely to become unreliable for long-term failure predictions (Lane et al., 1986). The omission of the time dimension will then limit the usefulness of the model when financial institutions often are interested in the ability of a predicting more than one year ahead at the time. With the use of annual accounts two, three or four years prior to bankruptcy, the model are considered to have some ability to predict whether a company will go bankrupt or not in the years, $t + 2$, $t + 3$ or $t + 4$ (Deakin, 1972).

Finally, as previously outlined, the failure of a company is not a sudden or unexpected event and the fact that classic statistical models do treat company failure as a steady state rather than a process will result in serious drawbacks (Laitinen, 1993; Laitinen & Kankaanpaa, 1999). Company failure is a result of bad performance over a period of time and can be seen as a failure process of different phases, where each phase is characterized by a specific development of the variables. The relative importance of each prediction variable for the detection of bankruptcy is then not constant over time (Daubie & Meskens, 2002). Moreover, in practice there are a wide variety of failure paths which classic statistical failure prediction models do not consider possible and this may cause serious consequences (Laitinen et al., 2014). This because the relative importance of the variables and the accuracy of the predictions are dependent by the frequency of occurrence of both the different kinds of failure paths and the different phases of the failure process that are in the sample of failing firms.

2.4.1 Sequential data

Since bankruptcy is not a static event, we need to consider the time dimension when modeling bankruptcy prediction. Usually this is done by splitting the in-sample and out-of-sample data by years, where the model is trained on a set of prior accounting years, validated on the next years, and tested on a set of withheld accounting years, usually the last couple of data periods (k-fold cross validation for instance). However, many bankruptcy prediction models use one-period financial statements to avoid model complexity, even though financial failure are not generally caused by one bad year, but as mentioned previously, usually the result of inadequate decision making over several periods (Campbell et al., 2008; Kim et al., 2020). A solution to this is utilizing sequential accounting data. This enables the model to understand the development of a company and its financial situation over time. Additionally, this reduces the chance of the model classifying a company suffering from a temporary bad situation as bankrupt. Therefore, Kim et al. (2020) suggest similarly to Shumway (2001), that multi-period sequential models are more appropriate for bankruptcy prediction, and mentions RNNs as an example. However, complex models able to process sequential data such as RNNs, often suffer from the problems related to interpretability presented in Section 2.2.2.

2.5 Previous work

In this section, previous work in regards the theoretical background is presented. This includes an overview of literature concerning machine learning and deep learning methods for bankruptcy prediction, before we further introduce previous research using different strategies to account for the black box problem and the imbalanced dataset problem. Finally, previous literature utilizing time sensitive modeling is presented.

2.5.1 Bankruptcy prediction methods

Researchers have utilized and researched several different statistical models and machine learning techniques for bankruptcy prediction over the years. However, as this thesis concerns the use of deep neural networks for bankruptcy prediction, no extensive description will be given regarding other methods. Still, we give a brief overview of research into the most used statistical and machine learning methods, as this relates to the adoption of deep neural networks in practice. This overview is followed by previous work in regards to deep learning for bankruptcy prediction.

Statistical and machine learning methods

Going back more than 50 years discriminant analysis (Altman, 1968) and logistic regression (Ohlson, 1980) were utilized to predict bankruptcy. Even neural network models for bankruptcy prediction came to fruition as early as the 90s (Odom & Sharda, 1990). According to Shi and Li (2019b) these are still the most researched methods.

However, many other models have also been researched. As for the statistical models, Shumway (2001) forecasted bankruptcy using a hazard model. He argued that static one-period models were inappropriate for bankruptcy prediction due to the fact that they do not take the time dimension into consideration. Multivariate discriminant analysis have also been a popular choice for forecasting bankruptcy.

When it comes to machine learning and artificial intelligence models, Support vector machines, decision trees and rough sets have also been popular (Alaka et al., 2018; Shi & Li, 2019b). Other models such as Adaboost, k-nearest neighbors and random forests have also been subject for investigation and experimentation for bankruptcy prediction (Shi & Li, 2019b).

Lastly, Norges Bank have since 2001 been using the SEBRA-model (Bernhardsen, 2007) to predict bankruptcies among Norwegian corporations, where they use a general additive modeling method. This is a linear model where the target variable is the sum of non-linear combinations of independent variables. A in-depth description of the model can be found in Eklund et al. (2001).

Deep learning methods

As stated by Qu et al. (2019) there has been a lack of research into the application of deep learning methods in finance and management. Still, some deep learning methods has been utilized for bankruptcy prediction in recent years. Alexandropoulos et al. (2019) created a deep fully connected feed-forward neural network for bankruptcy prediction with financial ratios as features. They compared the results to other methods such as logistic regression and found the deep learning model outperformed the rest based on AUC.

Hosaka (2019) employs a CNN for bankruptcy prediction. Firstly, the author calculates financial ratios and all combinations of correlation coefficients from financial statements. Thereafter an image is created using the financial ratios and a Monte Carlo simulation, that in turn can be analyzed by the CNN-model.

Recurrent neural networks (RNN) has also been put to use for bankruptcy prediction. The study of Kim et al. (2021) utilizes combined quarterly accounting and daily market data to test two RNNs, being traditional RNN and a LSTM network, against three benchmark methodologies: logistic regression, random forest, and support vector machines. The authors conclude, on the basis that the traditional RNN and LSTM network outperform the benchmark models with AUCs of respectively 0.7286 and 0.6707, that machine learning methodologies with the ability to pick up and process sequential data outperform models without this ability.

Moen (2020) constructed a traditional RNN network and a LSTM network trained on a previous version of the same dataset used in this thesis. The networks used a sequence of four accounting years, and 30 features. He found the networks to have good predictive performance, with the LSTM network performing slightly better than the traditional RNN network, with AUCs of 0.8836 and 0.8795 respectively. However, the LSTM network achieved a higher Brier score then the traditional RNN, with a score of 0.1295 compared to 0.1261, meaning the LSTM network was less confident in its predictions. He also found the traditional RNN and LSTM network to have higher predictive performance than logistic regression and tree-based models. His LSTM network consisted of one LSTM layer and two fully connected layers, all with 10 nodes. He found no substantial improvements when exploring more advanced architectures.

Other studies of RNNs for bankruptcy include the work of Jang et al. (2021) and Vochozka et al. (2020). The former focused on bankruptcy prediction of construction contractors. The three models that predicted bankruptcy within one, two and three years achieved an accuracy of 98%, 95.3% and 93%. The latter predicted the future development of manufacturing companies.

A combination of RNN and CNN, recurrent convolutional neural network (R-CNN) has also been used to predict bankruptcy (Becerra-Vicario et al., 2020). They use a deep R-CNN model to predict bankruptcy over a three year period in the restaurant industry. This method was proven to be effective on time series data.

Though this literature have generally found deep learning methods to have high predictive performance, the models often lack transparency into the decision process. For instance, Kim et al. (2021) expresses concern that his RNN and LSTM network cannot clearly indicate the importance of each individual explanatory variable as a consequence of model complexity and the black box problem. Therefore, to apply such models in practice, this issue needs to be addressed.

2.5.2 Interpretability of bankruptcy prediction models

Some methods to increase interpretability of machine learning models for bankruptcy prediction have been utilized in previous literature. These are mainly the LIME and the SHAP framework briefly described in Section 2.2.3.

Park et al. (2021) utilized LIME to explain the feature importance for each data point of their XGBoost and LightGBM models. They found that feature importance can be meaningfully extracted by using LIME, and that the possibility of observing the important features can be used as a basis for choosing eligibility requirements, and the fair treatment of loan applications. The LIME algorithm was also found to be effective in explaining deep neural network, and provided complementary interpretability to the decision tree model in the study of Chou (2019).

SHAP has been used to increase interpretability of deep learning models for bankruptcy prediction by Moen (2020) in his master thesis. Further, Jang et al. (2021) also used SHAP to interpret and measure feature impact of their LSTM-model for predicting bankruptcy in the constructor industry. Moreover, SHAP was utilized by Schalck and Yankol-Schalck (2021) to extract both local and global explanations of their XGBoost model for bankruptcy prediction amongst French SMEs. They state that their findings support practical applications for managers, financial institutions, and policy makers where the SHAP framework sucessfully increased model interpretability.

Though, SHAP and LIME are some of the most used methods to interpret machine learning models, are their use in bankruptcy prediction literature still scarce. However, they have been successfully utilized in other fields of research. Park and Yang (2022) found SHAP to be a helpful tool to interpret the prediction of growth rates and economic crisis, while Parsa et al. (2020) used the SHAP framework for both global and local explanations when predicting the probability of car accidents on highways. Further, SHAP has also been utilized in the field of medicine, Janizek et al. (2018) used the framework to interpret their tree-based models for predicting optimal drug combination to treat cancer patients. Additionally, Yang et al. (2022) used SHAP for increased interpretability of a LSTM network for forecasting tuberculosis incidents.

2.5.3 Feature effects on bankruptcy risk

There is no shortage of previous research regarding accounting-based predictor variables and their effects on bankruptcy risk. Still, no extensive overview of this literature will be presented in this section. However, to evaluate the trustworthiness of the SHAP framework, we need to compare our findings to previous work. Therefore, this section provides previously found feature effects used for comparisons in Section 6.2.

Firstly, Cultrera and Brédart (2016) found lower values of financial ratios such as profitability and liquidity increased bankruptcy probability, but did not find significant impact on model prediction from their solvency variable. However, Brîndescu (2016) concludes that companies with higher solvency have lower bankruptcy risk.

Dielman and Oppenheimer (1984) notes that changes in dividends policy can say a lot about a firm's prospects, indicating what the company leadership believes in regard to the financial situation. Specifically, a reduction of dividend payout may be an indicator of financial distress. Murekefu (2012) further found a strong positive relationship between company performance and dividend payout, noting that one of the major factors influencing dividend policy is profitability. This is backed by Kanakriyah (2020) who found that dividend policy had significant impact on company performance, and claim that dividend-payout ratio and dividend yield had significant influence on company performance prediction.

Salim and Yadav (2012) finds that a high ratio of total debt to total assets has a negative impact on performance measurements such as return on assets and return on equity. This is supported by Modina and Pietrovito (2014) who further found that high level of debt, limited supply of capital and high interest expenses is associated with higher risk of bankruptcy. However, Ogachi et al. (2020) found the opposite relation between debt ratio and bankruptcy probability. This illustrates the problem stated by Filipe et al. (2016), that feature effects often differs between data.

2.5.4 Data balancing

In this section, literature utilizing different data balancing strategies and methods are presented. First, previous research using resampling methods are presented, followed by cost-sensitive learning, and lastly, feature selection.

Resampling methods

In the case of under-sampling for bankruptcy prediction using deep neural networks, both Moen (2020) and Pelja and Wahlstrøm (2021) used matched under-sampling to obtain data with an equal amount of bankrupt and non-bankrupt observations, with the apparent drawbacks of wasting data and producing strong biases through the data distribution shift (Moen, 2020).

Over-sampling techniques have also been popular to address data imbalance in deep neural networks for bankruptcy prediction. Notably, the resampling techniques synthetic minority oversampling technique (SMOTE) and adaptive synthetic sampling approach (ADASYN) have been applied in the literature (Aljawazneh et al., 2021; Jang et al., 2021; Kim et al., 2021). SMOTE creates artificial data points based on the original data. Similarly, ADASYN also generate synthetic data, but also consider the distribution of the original data points. Aljawazneh et al. (2021) also tested and compared the hybrid approaches SMOTE-tomek and SMOTE-ENN with other SMOTE variants. They conclude that SMOTE-ENN proved the mutual superior balancing technique according to their chosen metrics.

Cost-sensitive learning

Different cost-sensitive learning strategies have been tried out to combat the imbalanced dataset problem for bankruptcy prediction, though it is much less popular than resampling methods. Chen et al. (2011) proposes an evolutionary algorithm approach to cost-sensitive bankruptcy prediction using a neural network, proving the method effective on real-life data.

Moreover, Ghatasheh et al. (2020) utilized cost-sensitive ensemble methods for bankruptcy prediction. The ensemble methods include AdaBoost, Bagging and random forest. They conclude that the random forest algorithms utilizing the proposed cost-sensitive learning strategy achieved competitive predictive performance.

Le et al. (2019) state that both oversampling and cost-sensitive learning techniques are viable options to deal with the class imbalance issue, and improves predictive performance of bankruptcy prediction models. However, they further state that a combination of the approaches produces even better results, and proposes a hybrid approach combining SMOTE-ENN and a cluster-based boosting algorithm. They conclude that the method outperform other existing balancing strategies for bankruptcy prediction on the Korean market.

To the best of our knowledge, a cost-sensitive learning method have not exclusively been used to address the imbalanced data problem for deep neural networks for bankruptcy prediction.

Feature selection

Some methods have been used for feature selection in the bankruptcy prediction literature. Paraschiv et al. (2021) used a wrapper method, that created new subsets by sequentially adding and removing features in order to evaluate their fit to the model. Further, Kou et al. (2021) utilized a two-stage multiobjective feature selection method, that selects the top k features by their relevance, before the method, much like the wrapper method of Paraschiv et al. (2021), finds an optimal feature subset that optimizes model performance.

In recent years, SHAP and Shapley values have been used for feature selection, with the added benefit of being more interpretable than other methods (Fryer et al., 2021; Marcílio & Eler, 2020; Xiaomao et al., 2019). In the domain of bankruptcy prediction, Xiaomao et al. (2019) found that the method discovered all influential features, and further state that the SHAP framework performed just as well as other more common feature selection methods. Likewise, Marcílio and Eler (2020) conclude that from their experiments, the SHAP framework for feature selection proved superior to other methods. However, Fryer et al. (2021) questions this use of the SHAP framework, and state that the properties of SHAP "do not *in general* provide any guarantee that the Shapley value is suited to feature selection, and may, in some cases, imply the opposite". A description of the SHAP properties can be found in Section 3.2.1.

2.5.5 Time sensitive modeling for bankruptcy prediction

In this section we present previous work regarding machine learning incorporating the time dimension for bankruptcy prediction. In terms of deep neural networks, RNNs and LSTM networks are able to utilize sequential data for bankruptcy prediction. Kim et al. (2021) incorporates sequential data for their RNN and LSTM network, leading to better predictive performance than their benchmark methods. Moreover, Moen (2020) uses sequential accounting data to predict bankruptcy using both a traditional RNN and a LSTM network. Others have also tried incorporating time-series variables to account for the issue of bankruptcy not being a steady state. This include Campbell et al. (2008) who explored the time-series variation of variables for bankruptcy prediction, and Duan et al. (2012) who created a forward intensity model for bankruptcy prediction.

In spite of the problems discussed in Section 2.4, studies incorporating sequential data into machine learning for bankruptcy prediction remain rather scarce (Kim et al., 2021). In this thesis, to not neglect the time dimension in bankruptcy prediction, we use LSTM networks to process sequential accounting data to predict bankruptcy probabilities.

2.6 Summary of theoretical background and previous work

In this chapter we have presented the theoretical background and previous work regarding bankruptcy prediction. First, bankruptcy prediction is an important topic for company stakeholders, wanting to use bankruptcy prediction models for better decision making. Deep learning methods have been proven to have superior predictive performance compared to traditional machine learning methods, but are often criticized for their lack of interpretability, which have reduced their use for bankruptcy prediction in practice. Furthermore, bankruptcy data often suffers from imbalanced class distributions, as bankruptcy is a rare event. Therefore, to train bankruptcy prediction models, it is often necessary to employ data balancing strategies that introduce significant bias to the model. Moreover, simpler machine learning methods often neglect the time dimension in bankruptcy prediction, and treats bankruptcy as a steady state. Therefore, Kim et al. (2020) argues that models capable of processing sequential data, such as RNNs, are more appropriate for bankruptcy prediction.

Chapter 3

Methods for bankruptcy prediction and model evaluation

As presented in section 2.5.1, several statistical and machine learning methods have been used for bankruptcy prediction. This chapter will describe the background of the chosen models, encompassing an introduction to neural networks, RNNs and LSTM networks. Furthermore, methods for increased model interpretability is presented. Lastly, we describe the evaluation metrics AUC and Brier score, and introduce a cost-sensitive learning strategy. Note that all figures are created by the authors if not otherwise specified.

3.1 Deep learning methods

This section encompasses the deep learning methods utilized in this thesis. First, an introduction to neural networks and its aspects are presented. Thereafter follows a description of RNNs and LSTM networks.

3.1.1 Neural networks

Neural networks are machine learning models partly conceptualized by drawing inspiration from our understanding of the brain (Chollet, 2018). As mentioned in Section 2.1.3 the most basic form of neural networks are fully connected feed-forward neural networks. The structure consists of three types of layers of interconnected computational neurons, being the input layer, one or more hidden layers, and an output layer. Each connection is given a weight creating a web of weighted information flow between the nodes of each layer. When an input node is activated, it transmits its information to the connected nodes in the next layer, where the receiving nodes sums up the weighted inputs linearly, before applying a non-linear transformation to create its own output to be further transmitted to the next layer of neurons (Laitinen & Kankaanpaa, 1999). This process repeats until the information has arrived in the output layer where the final weighted summation and non-linear transformation takes place, creating the model prediction. The structure of a fully connected feed-forward neural network is depicted in Figure 3.1.

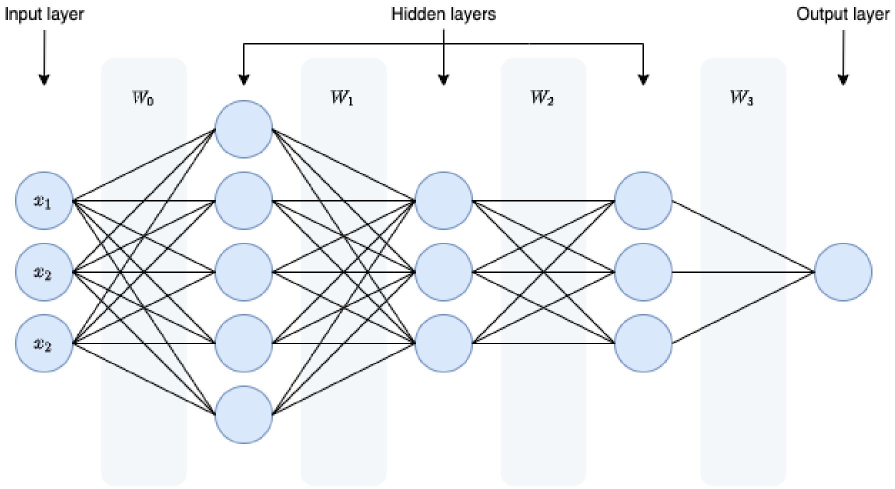


Figure 3.1: The structure of a fully connected feed-forward neural network with three hidden layers of five, three and three nodes respectively.

The model prediction is then compared to the actual target value through a loss function that computes a score describing how far off the prediction was from the true value. This score is fed through an optimizer which in turn implements the backpropagation algorithm. This algorithm assigns adjustments to the weights of each connection to reduce the loss score by making the model generate predictions that are closer to the target value. Repeating this process results in model training, where the weights are adjusted to minimize the loss function. This cycle is depicted in Figure 3.2.

3.1.2 Loss function

The loss function (or the objective function) is the quantity to be minimized (or maximized) during training. Therefore, choosing the correct loss function is essential when training neural networks as it represents the degree of success for the chosen task. Consequently, if the loss function does not fully correlate with the success of the task, the model may end up doing undesirable things (Chollet, 2018).

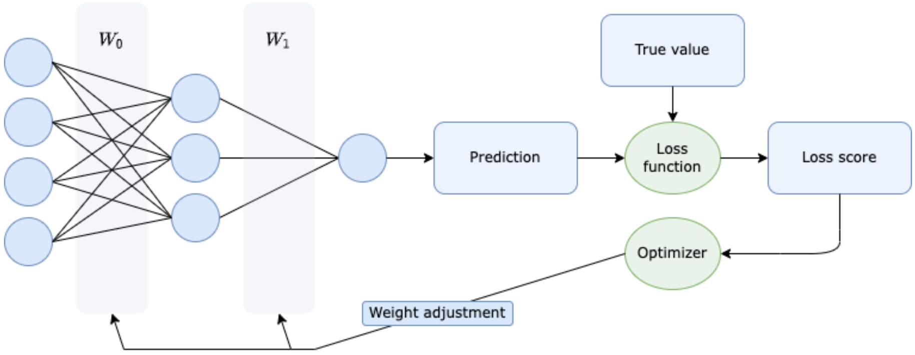


Figure 3.2: The training process of neural networks. The output of the neural network is compared to the true value of the observation, before calculating the loss score. Thereafter, the weights are adjusted to decrease the loss score of the next iteration.

When it comes to problems like classification tasks, by far the most used loss function is binary cross-entropy. This loss function compares each of the predicted output probabilities to the true class value, and subsequently gives the model a score (log loss) based on how well it has predicted:

$$L = \frac{1}{N}\sum_{i=1}^{N} -(y_i \log(p_i) + (1 - y_i)\log(1 - p_i)) \tag{3.1}$$

where N is the number of data points, y_i is class 1 (bankrupt), and p_i is the probability of class 1.

Though binary cross-entropy is the most used loss function for binary classification, the method carries significant drawbacks when it comes to real world classification problems. As discussed in Section 2.3, arithmetic accuracy metrics such as cross entropy can be misleading when the class distribution is imbalanced, as is typically when it comes to bankruptcy prediction. Yan et al. (2003) proposes a solution by maximizing AUC directly through an approximation of the Wilcoxon-Mann-Whitney statistic. Still, as we could not find a pre-existing implementation of this loss function for Tensorflow or Keras, and considered an implementation from scratch outside the scope of the thesis, binary cross-entropy was used. This despite the fact that this function does not necessarily fully correlate to the success of the task.

3.1.3 Optimizer

The optimizer determines how the network weights are to be updated based on the loss function through a method called stochastic gradient descent (SGD) and back propagation (Chollet, 2018). Without going into to much detail, SGD is a probabilistic version of regular gradient descent, where probabilistic refer to the inclusion of some randomness. This randomness comes from the random batches of data used in the calculation of the gradient instead of including the entire dataset, thus greatly reducing the time it takes for the model to converge. Therefore, SGD is more suitable for large datasets. Though this random sampling is the methods greatest advantage, it also means greater variance.

Multiple built in optimizers can be found in the Keras library, such as RMSProp, AdaGrad, AdaMax and Adam. However, our focus will be on the Adam algorithm. The name comes from adaptive moment estimation, and was designed to fuse the advantages of AdaGrad and RMSProp, two other popular optimizer options. A description for the algorithm can be found in Kingma and Ba (2017).

3.1.4 Activation functions

As described in Section 3.1.1 a receiving node sums up weighted inputs, before adding non-linearity to create its own output. This is the job of the activation function, that additionally through this process also decides whether or not the neuron should be activated and thereby send information further into the network. There are many activation functions to choose from, but the most commonly used for binary classification tasks are *sigmoid* (σ) and hyperbolic tangent (*tanh*) as both transforms the output logistically. Still, rectified linear unit ($ReLU$) and *softmax* have also been incorporated for deep neural networks for bankruptcy prediction (Alexandropoulos et al., 2019; Kim et al., 2021).

The sigmoid function is a smooth differentiable approximation of a threshold unit, and compresses the inputs into the range $(0, 1)$. It is therefore widely used in output layers for binary classification tasks when the goal is not direct prediction, but rather to give probabilities for an observation being a specific class (Zhang et al., 2021).

$$\sigma(x) = \frac{1}{1 + \exp(-x)} \tag{3.2}$$

Similarly to the sigmoid function, the hyperbolic tangent compresses the inputs, though into the range $(-1, 1)$.

$$\tanh(x) = \frac{1 - \exp(-2x)}{1 + \exp(-2x)} \tag{3.3}$$

Without going into to much detail, the main reason why tanh often is preferred as activation functions for hidden layers is that tanh exhibit point symmetry around the origin, meaning it is more likely to produce outputs that on average is close to zero. This in turn often means faster learning and convergence (Buttou et al., 2012). This is the same argument as for normalization of data for neural networks.

3.1.5 Neural network challenges

Neural networks are known to be notoriously difficult to train (Hastie et al., 2009). This section provides a overview of the challenges in training deep neural networks. Firstly, deep neural networks are particularly data hungry. Even training a neural network for simple tasks often require large amounts of training data (Aggarwal, 2018). Additionally, the performance of deep neural networks highly depends on the chosen hyperparameters and architecture (Zhang et al., 2019). However, finding the correct hyperparameters and structure is challenging. Moreover, due to model complexity, neural networks have a tendency to overfit the data, trying to find the global minimum of the training set, rather than the best solution for the out-of-sample data. Several methods to combat this issue have been utilized, where early-stopping algorithms and dropout layers have been popular. Additionally, the loss function often have many local minima, meaning the starting weights have great impact on which minimum point the model finds. Input scaling also have significant impact on the model solution. Therefore, when optimizing neural networks, best practice is to standardize all inputs since this also determines the weight scaling (Hastie et al., 2009).

3.1.6 Recurrent neural networks

As discussed in Section 2.4, when predicting bankruptcy, the time distribution of the data needs to be addressed. This means including a temporal dimension to the data, and thereby assuming some relationship between previous data and future data. Traditional neural networks such as fully connected neural networks have no memory, and process each input independently (Chollet, 2018). In order to process sequential or temporal series of data, the entire sequence needs to be shown to the network at once. Consequently, the entire sequence has to be transformed into a single data point (vector). Alternatively, other types of neural networks can be utilized, like the RNNs used in this thesis.

RNNs introduces a *recurrent layer* into the model. This layer processes sequences of data by going through the elements and calculating a *hidden state* h_t containing information about what it has seen so far (Chollet, 2018). This state is calculated by the input of the current time step, together with the hidden state of the previous timestep h_{t-1}:

$$h_t = \phi(x_t W_{xh} + h_{t-1} W_{hh} + b_h) \tag{3.4}$$

where $x_t \in \mathbb{R}^{n \times d}$ is a mini batch of inputs at time step t with batch size n and d inputs, $W_{xh} \in \mathbb{R}^{d \times h}$ is the weight of the mini batches with h number of hidden units, $W_{hh} \in \mathbb{R}^{h \times h}$ is a weight describing how the new hidden state should use the information stored in the previous hidden state, $b_h \in \mathbb{R}^{1 \times h}$ is the bias, and ϕ is the activation function of the hidden layer. An illustration of the process of a RNN can be found in Figure 3.3

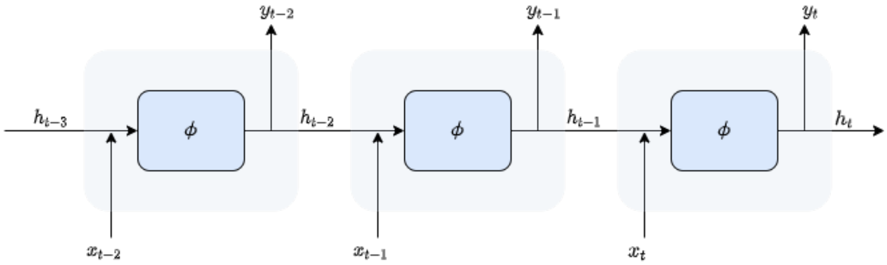


Figure 3.3: An illustration of how a RNN uses the previous hidden state to create the next hidden state, in addition to an output y of the current time step.

The biggest issue regarding RNNs is retaining long-term dependencies. Though RNNs do retain some information between time-steps, due to the vanishing gradient problem, inputs introduced to the model many time-steps before becomes small and consequently “forgotten” by the model.

3.1.7 Long short-term memory networks

Long short-term memory (LSTM) networks was introduced by Hochreiter and Schmidhuber (1997) as an answer to the vanishing gradient problem for RNN. LSTM networks therefore seek to retain long-term dependencies across time steps by introducing a memory cell c_t which together with the hidden state h_t control the flow of information. This section gives a conceptual description of LSTM networks inspired by Zhang et al. (2021). A deeper dive into the mathematics can be found in Hochreiter and Schmidhuber (1997).

The base idea is using the memory cell c_t to scale the hidden state h_t output at every time step t. The memory cell is controlled by three gates: the Forget gate $F_t \in \mathbb{R}^{n \times h}$, the Input gate $I_t \in \mathbb{R}^{n \times h}$ and the Output gate $O_t \in \mathbb{R}^{n \times h}$. The forget gate controls what information to remove form the cell state, whereas the Input gate determines both what and how much new information needs to be stored. The Output gate produces the cell output. Each gate has their own set of weights for both the previous hidden state h_{t-1} and the input x_t. The gate outputs at each time step t are linear combinations of the input, the previous hidden state, and their respective set of trainable weights W transformed by a sigmoid activation function σ resulting in values in the range $(0, 1)$. The gates are calculated through the following equations:

$$\begin{aligned} I_t &= \sigma(x_t W_{xi} + h_{t-1} W_{hi} + b_I), \\ F_t &= \sigma(x_t W_{xf} + h_{t-1} W_{hf} + b_F), \\ O_t &= \sigma(x_t W_{xo} + h_{t-1} W_{ho} + b_O) \end{aligned} \tag{3.5}$$

where $W_{xi}, W_{xf}, W_{xo} \in \mathbb{R}^{d\times h}$ are the weight parameters for the input x_t for each gate, $W_{hi}, W_{hf}, W_{ho} \in \mathbb{R}^{h\times h}$ are the weights for the previous hidden state for each gate, $b_I, b_F, b_O \in \mathbb{R}^{1\times h}$ are the biases, and σ is the sigmoid activation function.

Next, a candidate memory cell $\tilde{C}_t \in \mathbb{R}^{n\times h}$ creates a set of new candidate values to be added to the cell state. Though the computation is similar to the gate functions in (3.5), the linear combination is transformed through a hyperbolic tangent activation function, resulting in values in the range $(-1, 1)$.

$$\tilde{C}_t = \tanh(x_t W_{xc} + h_{t-1} W_{hc} + b_{\tilde{C}}) \tag{3.6}$$

To create the new cell state $c_t \in \mathbb{R}^{n\times h}$, we first multiply the old cell state c_{t-1} by F_t using element wise multiplication denoted $\odot$, removing undesired information from the previous state. Additionally, we add new information by multiplying the candidate values $\tilde{C}_t$ with the input gate I_t, resulting in the following equation:

$$c_t = F_t \odot c_{t-1} + I_t \odot \tilde{C}_t \tag{3.7}$$

Lastly, the new hidden state $h_t \in \mathbb{R}^{n\times h}$ is computed by multiplying the Output gate with the hyperbolic tangent of the new cell state ensuring that the values of h_t remains in the range $(-1, 1)$:

$$h_t = O_t \odot \tanh(c_t) \tag{3.8}$$

The whole process of the LSTM cell is illustrated in Figure 3.4.

Because of the LSTM network structure, the inputs are required to be sequences of data. Therefore, the input to the LSTM layers must be three-dimensional in the form of {sample, time-step, features}. *Sample* is the amount of observations sent into the LSTM network. *Time-step* is the sequence length of the observations. Lastly, *features* are the amount of features used to describe

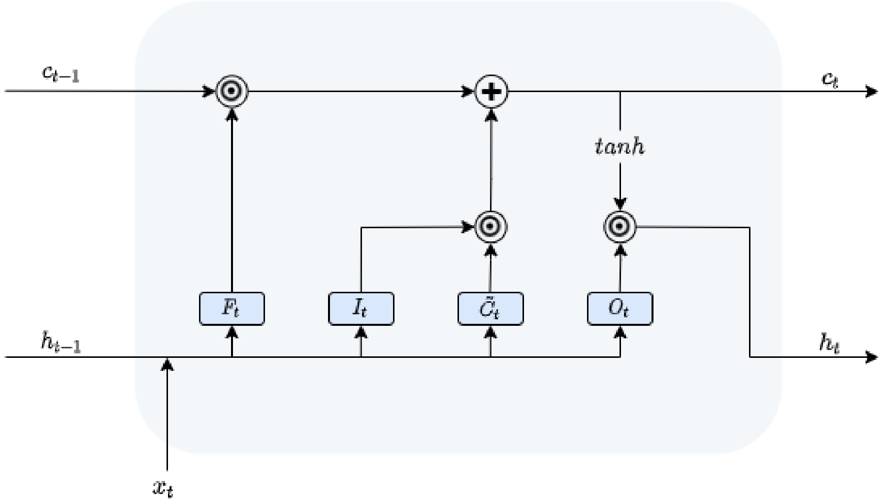


Figure 3.4: The LSTM cell, illustrating the process of how the cell state and hidden state are calculated.

the input data. Consequently, when validating and testing on out-of-sample data, they are also required to be of the same structure. This puts some restrictions on the training and test splitting that is discussed in 4.2. Still, this method enables the use of sequential data for bankruptcy prediction.

3.2 Methods for increased interpretability

The ability to correctly interpret a prediction model's output is essential as it builds user trust, supports understanding of the process being modeled, and provides insight into how a model may be improved (Lundberg & Lee, 2017). It also facilitates its use in practice as discussed in Section 2.2.2. The academic literature provides several methods for increased interpretability of complex black box models (LIME, DeepLIFT, Layer-Wise Relevance Propagation, Classic Shapley Value Estimation, SHAP). In this thesis the main focus will be the use of SHAP for interpreting deep learning bankruptcy prediction models. At the the center of SHAP we find Shapley Values.

3.2.1 Shapley values

The Shapley values stem from coalitions game theory, where the feature value of a data instance act as players in coalition (Molnar, 2022, Chapter 9.6). The core concept is to give each player (feature) a value representing how much they contribute to the expected gain, relative to what the expected gain would have been if the player did not participate. In other terms, the Shapley value is the average marginal contribution of a feature value across all possible coalitions, and indicate how to fairly distribute the prediction among the features (players) and consequently the impact each feature has on the output. The values therefore offer a way of observing the feature importance and effect of model outputs.

For the classic Shapley value estimation let all feature subsets $S \subseteq F$, where F is the set of all features. Let $f_{S\cup\{i\}}$ denote a model trained with the feature i present, while another model f_S is trained with the feature withheld. To evaluate the features effect, the predictions from both models are compared to the input on the feature subset. Additionally, since the effect of withholding a feature depends on the other features in the model, the comparison is done for all possible subsets, where the Shapley value is the weighted average of all the differences. The Shapley value is calculated from the following equation:

$$\phi_i = \sum_{S\subseteq F\setminus\{i\}} \frac{|S|!(|F|-|S|-1)!}{|F|!}[f_{S\cup\{i\}}(x_{S\cup\{i\}}) - f_S(x_S)] \tag{3.9}$$

where x_S represents the value of the input features in the set.

The Shapley value is the only feature attribution method that satisfy all properties that define a fair payout: efficiency, symmetry, dummy and additivity (Molnar, 2022, Chapter 9.5). Efficiency means that the contributions of all features must together add up to the difference of prediction and the average. The symmetry property describes that two feature contributions should be the same if their contribution across all possible coalitions is equal. Further, the dummy property state that if a feature does not change the predicted value, the Shapley value should be zero. Additivity means that when averaging all feature Shapley values across all individual coalitions you get the combined Shapley value of the model. Because Shapley values satisfy these properties, it has a strong theoretical foundation.

3.2.2 Shapley additive explanations

Shapley additive explanations (SHAP) was presented by Lundberg and Lee (2017) and a framework for explaining predictions of black box models. The base idea of SHAP is constructing a simplified explanation model defined as any interpretable approximation of the original model. SHAP specifies the explanation model g as:

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i \tag{3.10}$$

where $z' \in \{0, 1\}^M$ is the simplified inputs that map the original inputs. For a more extensive description see Lundberg and Lee (2017).

SHAP feature effect and importance

The calculated SHAP value represents a specific feature's influence on the model prediction. Therefore, we interpret the feature effect on model prediction by using the SHAP value. To interpret the local feature importance in SHAP, we observe the absolute Shapley value of a feature for the prediction. The greater the value, the higher feature importance. This enables interpretations of magnitude of impact for each feature on a specific prediction.

SHAP also allows for global interpretations, meaning the framework also gives insight into the general logic of the model. As Shapley values are locally calculated, we average the absolute values per feature across the data by means of equation 3.11, consequently creating global explanations of feature importance based on the theoretically solid foundation of Shapley values.

$$\frac{1}{n} I_j = \sum_{i=1}^{n} |\phi_j^{(i)}| \tag{3.11}$$

It is important to note that the Shapley value is not the average difference in predicted value after removing that feature from the model. The interpretation is rather the contribution of a feature value to the difference between the actual prediction and the average prediction, given the current set of feature values (Molnar, 2022, Chapter 9.6).

3.2.3 Deep SHAP

Deep SHAP is an approximation of SHAP values used for interpreting deep models and are included in the Python SHAP-package. Deep SHAP transforms SHAP values for smaller parts of the network into values for the entire network. The mathematical and theoretical justification for this can be found in Lundberg and Lee (2017). However, such an algorithm can be computationally intensive based on the number of data samples used. Therefore, even though there are no rule of thumb regarding the sample size (Molnar, 2022, Chapter 9.6), the deep SHAP documentation state that a sample size of 1000 is “a very good estimate of the expected values” (Lundberg, 2018a).

3.2.4 Disadvantages of SHAP

Though SHAP has many advantages over other methods with similar goals, it also comes with some disadvantages. Firstly, the calculation of Shapley values and consequently SHAP is slow. This comes from the substantial amount of possible coalitions of features when the feature set is large. Therefore, for real-world problems only approximations of Shapley values can be calculated in a feasable amount of time. This is done by sampling coalitions and limiting the number of iterations (Molnar, 2022, Chapter 9.6).

Another disadvantage is demonstrated by Slack et al. (2020). They claim that it is possible to create intentionally misleading interpretations based on SHAP. This can severely impact the applicability of SHAP as a model interpreter for complex bankruptcy prediction models in the real world.

3.3 Model evaluation

In machine and deep learning, performance measures are essential to be able to evaluate and compare the prediction models overall quality and performance. This is typically done by utilizing a set of evaluation metrics, composed of single score values, making it easy and intuitive to compare model performance. As mentioned in Section 2.3, arithmetic accuracy metrics is unfit for bankruptcy prediction when the data is highly imbalanced. Another popular metric is F-score. However, this requires specifying a predefined threshold value for when a prediction is bankrupt or not. As we want to interpret the model output as probability of bankruptcy, this is not a good fit for our purpose. Still, evaluation metrics such as AUC and Brier score do not require a threshold value, therefore being a better fit for our thesis.

3.3.1 AUC

In binary classification problems, the evaluation and performance of classifiers are often measured using the *Area under the receiver operator characteristic* curve (AUC). The AUC of a classification function f expresses how good f is capable of distinguishing between classes or that the probability for a randomly selected positive example gets a higher score by f than a randomly selected negative example. The higher the AUC score is, the better f is at predicting the classes correctly.

The *receiver operator characteristic* (ROC) curve for a binary classification problem plots sensitivity (the true positive rate) as a function of 1−specificity (the false positive rate). By setting a decision threshold in the range from $(0, 1)$, the output of the predictive function f can be translated into a binary classification making up the ROC curve. If the predicted output is below the threshold, the prediction is 0, otherwise, 1 is predicted. Depending on the threshold, there is a trade-off between sensitivity (failed firms that have been correctly classified) and specificity (healthy firms that have been correctly classified). If the threshold is low, sensitivity of the 1-class is high and specificity is low. On the other hand, if the threshold is high, sensitivity is low, while specificity is high (He & Garcia, 2009).

$$\text{Sensitivity} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{3.12}$$

$$1 - \text{Specificity} = \frac{\text{False Positive}}{\text{False Positive} + \text{True Negative}} \tag{3.13}$$

For a random classification the ROC curve will have a linear slope stretching from origin to $(1, 1)$ and an AUC of 0.5, implying that a AUC above this is an improvement of pure guessing. The ROC curve is depicted in Figure 3.5 with 1−specificity on the $x-$axis and sensitivity on the $y-$axis. The AUC is then calculated as the total area under the curve, and is in the range of $(0, 1)$.

3.3.2 Brier score

A popular metric to evaluate the overall goodness-of-fit of binary and categorical values is the Brier score. It is a measure of the calibration of a set of probabilistic predictions, meaning how well the estimated probabilities of failure match the actual failure/non-failure observation. It is calculated as the mean squared error between each binary outcome and its predicted probability (Fenlon et al., 2018).

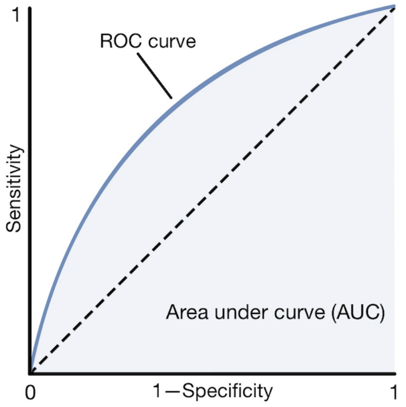


Figure 3.5: An illustrative ROC curve with an AUC greater than 0.5. The x-axis is $1-$Specificity (false positive rate), and the y-axis is sensitivity (true positive rate).

$$\text{Brier score} = \frac{\sum_{i=1}^{N}(Y_i - p_i)^2}{N} \tag{3.14}$$

where N is the total number of observations in the test set, Y_i is the actual outcome of record i (0 or 1), and p_i is the predicted probability of record i.

The Brier scores lies in the range $(1, 0)$, were a perfect model will receive a score of 0, while a model that keeps predicting probabilities close to 0.5 will receive a high Brier score. However, even with a high Brier score, the model can correctly classify all instances if the prediction probabilities are at the correct side of the threshold value. Therefore, the Brier score is useful to measure how confident the model is in making its probability estimates. This is still true even when no threshold value is specified.

3.4 Cost-sensitive learning

Cost-sensitive learning is a strategy for dealing with class imbalance by assuming a higher cost of misclassification for the minority class. However, as the true cost of misclassification is hard to find, and can not necessarily be given by an expert, the problem is finding a feasible value for the cost. Though some solutions have been suggested in the literature, no definitive answer to this problem for bankruptcy prediction have been found. Still, we will base our classification cost calculation on the suggestions from the Tensorflow web page “Classification on imbalanced data” (2022) as this method proved effective.

By changing the class weights in the neural network we can make the model pay more attention to correctly classifying a specific class:

$$\begin{aligned}\text{Weight}_0 &= \frac{1}{\text{Obs}_0} * \frac{\text{Total}}{2} \\ \text{Weight}_1 &= \frac{1}{\text{Obs}_1} * \frac{\text{Total}}{2}\end{aligned} \tag{3.15}$$

where Weight_0 denotes the weight of non-bankruptcy, while Weight_1 is the weight of a bankruptcy observation. Obs_x denotes the amount of observations for the specific class, and Total is the total number of observations. Note that this calculation is done based on the training set individually for each model and results in a classification cost similarly to having a 50/50 balance in the datasets.

Chapter 4

Method

In this section, we describe the chosen method of this thesis. To begin with, we present the data and the preprocessing. Further, we describe the splitting of training, validation and testset, before detailing the data balancing strategy. Thereafter, the implementation of the LSTM networks and the baseline neural networks are described. Finally, the model evaluation metrics and the implementation of the SHAP framework are presented.

4.1 Data

This thesis utilize a dataset of all unconsolidated annual financial statements of Norwegian private and public liability companies from the accounting years 2006 – 2019, as described by Wahlstrøm (2022). We will further in this section introduce our target variable and features, before describing the data preprocessing.

4.1.1 Target variable

To determine whether a company went bankrupt or not we used the dichotomous variable $\texttt{bankrupt}_{fs}$ as our target variable. The bankruptcy target variable is one if the financial statement is the last financial statement of the underlying company and the company has filed for bankruptcy in accordance to the variable `konkursdato`. Otherwise it is categorized as non-bankrupt, i.e. zero (Wahlstrøm, 2022).

4.1.2 Features

Our data set consisted of 281 accounting items from the companies' financial statements as well as some other company related data. The full list of accounting items are available in Wahlstrøm (2022). As the foundation for our bankruptcy prediction models, we used a set of 156 accounting based features, assembled by (Paraschiv et al., 2021) based on the original accounting items. The full list of features can be found in Appendix D.

Of our 156 input variables, only five were non-ratios. Three were log transformed accounting items being `age in years`, `total assets` and `financial expenses`, while the other two were dummy variables: `one if paid-in equity is less than total equity` and `one if total liability exceeds total assets`.

4.1.3 Data exclusion

In accordance with the project task, only small and medium-sized companies (SME) were included in the dataset. SMEs are defined in the EU recommendation 2003/361 either by staff head count, total turnover or balance sheet total. In line with the EU, Paraschiv et al. (2021) and Moen (2020), we define a company as SME if total turnover does not exceed EUR 50 millions, or total assets does not exceed EUR 43 millions. Additionally, the lower bound total assets was set to EUR 2 millions to exclude micro companies, with the added benefit of reducing data error and outlier problems, of which accounting data from small firms are susceptible (Paraschiv et al., 2021). Moreover, we only include public and private companies with the organizational forms ASA or AS, as they are required to report their financial statements to the Norwegian authorities, ensuring better data quality. We also excluded companies in the industries of "Financial and insurance activities", "Real estate activities", "Electricity and gas supply", and "Water supply, sewage, waste", following the prior literature (Mansi et al., 2010; Moen, 2020; Paraschiv et al., 2021).

4.1.4 Missing values and outliers

The accounting data used for creating the financial ratios do not contain any missing values, as missing values only mean that no value for the specific accounting item was given (Wahlstrøm, 2022). Therefore, the value is zero. To reduce the effects of outliers or recording errors, we take inspiration from previous literature (Chava & Jarrow, 2004; Moen, 2020; Paraschiv et al., 2021; Shumway, 2001; Tian & Yu, 2017; Tian et al., 2015) and winsorize all accounting based financial ratios on the 5th and 95th percentile. This means changing the value of each outlier to that of the nearest inlier, in our case the 5th or 95th percentile value for each feature. As our original feature set is large, many observations may be outliers in just a small amount of features, meaning a trimming scheme may discard good observations based solely on one feature being an outlier. Winsorizing was therefore chosen as the preferred method for protection against outliers, as it does not discard data.

4.1.5 Feature scaling

The range of values for the features in bankruptcy data can vary a lot depending on the company in question. Prediction methods, including neural networks requires the feature values to be normalized in order to feed the network with data ranging in the same interval for each input node (Angelini et al., 2008). To do this, a common approach is to use *min-max scaling*, monotonically transforming the feature values into a given range of $(0, 1)$, where the highest feature value become one, and the lowest become zero (Bao et al., 2019).

4.1.6 Data structure

The inclusion of sequential data in the networks meant the data had to be restructured in such a way that it consisted of sequences of feature values for X amount of accounting years (time-steps) for individual companies. This also meant introducing some considerable restrictions on our data. Firstly, we required that each company had X amount of accounting years (depending on the sequence length) included in the data, meaning that for the models requiring four sequential accounting years, companies that only sent inn three annual accounts was not included. This also meant that potential bankruptcy observations of companies that started to report their financial statements to the Norwegian authorities 3 years or less prior to a bankruptcy was lost. The implications of this for the training and test splitting is described in Section 4.2.

4.1.7 Feature correlation

As discussed in Section 4.1.1, we started with a set of 156 features. However, as a consequence of the exhaustive list of features, some were highly correlated. Still, as mentioned in Section 2.1.3, deep learning methodologies can recognize high-level nuances and patterns, meaning a features can have valuable input on the model prediction even with high levels of correlation. However, all fully correlated features should be removed before model training. Still, we also need to consider the nature of accounting data, and that even an accounting error of 1 NOK will result in the Pearson correlation coefficient not being exactly ±1. We calculated the correlation coefficient for all possible pairs of features, before removing all features with at least one coefficient of ±0.99. This resulted in a feature set consisting of 144 variables.

4.1.8 SHAP for feature selection

Feature selection is considered an important step of bankruptcy prediction, to avoid using redundant and irrelevant variables. We selected a subset of 30 features from our set of 144 variables using the SHAP framework. This was based on the top 30 features ordered by the feature contribution to the model prediction. The choice of 30 features was motivated by Paraschiv et al. (2021) who found model performance to even out as the number of features start to approach 30. Still, as stated in 2.5.4, the properties of Shapley values do not guarantee its suitability for a feature selection tool. Still, we utilized the method as it is increasingly prevalent in the literature, and we found the approach feasible for out purpose.

4.1.9 Key assumptions

Throughout this thesis, some assumptions regarding the data of the financial statements are made. First of all, we assume that the financial statements in our dataset are reported correctly, which may not always be true (e.g., because of entry errors or fraud). Further, we do not take changes in macro economical and other external factors into consideration, assuming homogeneity over time which may be unrealistic, especially as our time period included the financial crisis of 2007–2008. Additionally, when the denominator for a financial ratio is zero, we assume the value of the feature to be zero in accordance with (Paraschiv et al., 2021).

4.2 The splitting of training and test set

When it comes to bankruptcy prediction, the training, validation and test splitting scheme needs to consider the temporally distributed data to avoid time-leakage. This means training on previous data to predict the future. Methods such as k-fold cross validation enables such considerations. However, such a training scheme was deemed challenging due to the data structure described in Section 4.1.6, as it restricted the data too much when sequences of accounting data was created.

The solution was splitting the training, validation and test set by organization, while also keeping the accounting years partly separated. Concretely, the training set received 52.5% of the organizations, the validation set 17.5%, and the test set the last 30%. Further, the training and validation sets received the observations for their respective organizations from the accounting years 2006–2015, while the test set received data from 2015–2019. Though this means that 2015 is a common accounting year for both the training and test set, because of the data structure and the use of sequential data, this was deemed feasible. Please note that such a splitting scheme has to the best of our knowledge not been done before. This solution is depicted in Figure 4.1.

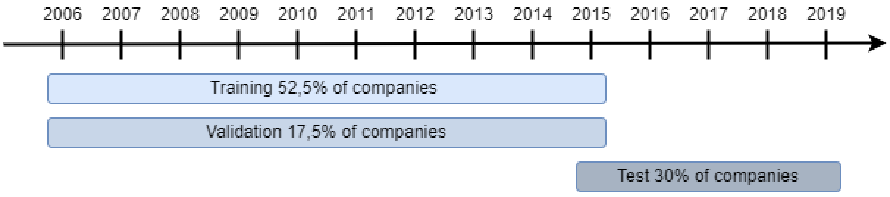


Figure 4.1: A visualization of the training, validation and test set split. Note that the training and model validation happened in the same time-period, while the testing was kept separate

Since we create networks able to process different lengths of sequences, we also had to create four groups of training, validation and test sets. These groups contain data of different sequence lengths ranging from one accounting year, to four. This results in differing amounts of observations between the sets of each group, as depicted in Table 4.1. Consequently, this also resulted in varying amounts of bankruptcy observations in each group and set, presented in Table 4.2. Note that the group names depicts the sequence lengths, and that 1-year do not contain a sequence of accounting data.

Table 4.1:
The amount of observations for the training, validation and test sets for each group

Observations	4-years	3-years	2-years	1-year
Train	33763	42761	53725	84032
Validation	11218	14410	18045	27994
Test	6116	10271	15548	25375

Table 4.2: The amount of bankruptcy observations for each training, validation and test set. Note that the name indicate the sequence length of the data

Bankrupt	4-years	3-years	2-years	1-year
Train	126	186	251	502
Validation	56	60	90	180
Test	24	34	61	121

4.3 Data balancing

As discussed, strategies to overcome the imbalanced dataset issue often comes with many drawbacks, especially in regards to increased model bias (Zmijewski, 1984). This is also true for cost-sensitive learning (Vo et al., 2019). Though resampling methods are common for deep neural networks for bankruptcy prediction, to our knowledge, the previous literature has not exclusively utilized a cost-sensitive learning strategy to deal with this issue in the domain. Still, this method allows for training on more realistic data compared to resampling methods, suggesting a better fit for our purpose. Therefore, we explored and used a cost-sensitive learning strategy to overcome the data imbalance in this thesis based on the method presented in 3.4.

Individual class weights were calculated for dealing with the class imbalance issue for each of the training sets as the number of bankrupt and non-bankrupt observations differed between the sets. These weights were calculated by means of equation 3.15 and are depicted in Table 4.3.

Table 4.3: The class weights used for training each model using a specific training set. Note that the name indicate the sequence length of the data

Class weight	4-years	3-years	2-years	1-year
Non-bankrupt	0.5019	0.5022	0.5023	0.5030
Bankrupt	133.9762	114.9489	107.0219	83.6972

The differences in class weights stem from the variance in class distribution between the training sets for each group of sets depicted in Table 4.1 and Table 4.2.

4.4 Neural networks implementation

This section encompasses how the neural networks utilized in this thesis was implemented. We start by describing the implementation of the LSTM networks, before we present how the baseline neural networks used for comparison purposes were implemented.

4.4.1 LSTM implementation

The use of a recurrent neural networks was motivated by the addition of the time dimension to the model, and more specifically its ability to process sequential data. Moreover, LSTM was the preferred choice motivated by the challenges of traditional RNNs (Section 3.1.6) and the recommendations of Chollet (2018).

As previously outlined in Section 3.1.5 the performance of neural networks greatly depend on the architecture and hyperparameters. When constructing the models for this thesis, inspiration was taken from multiple sources, mainly Chollet (2018), Zhang et al. (2021) and previous work on deep LSTM networks for bankruptcy prediction (Jang et al., 2021; Kim et al., 2021; Moen, 2020; Vochozka et al., 2020). All neural networks was implemented using the Keras library in Python with the default parameters if not otherwise specified.

In order to consider the impact of including longer sequences of accounting data, four separate LSTM networks were created. The first utilized a sequence of four accounting years and all 144 features (after correcting for correlations), and was used for feature selection. The last three LSTM networks all used the feature subset created from feature selection, while processing different sequences lengths of accounting data, being four, three and two years respectively. The networks were trained on the groups of training, validation and test sets with the corresponding sequence length.

All LSTM networks had the same basic architecture, consisting of an input layer, two hidden LSTM layers, a flatten layer, one dense layer, followed by a dropout and an output layer. As for the nodes in each layer, the input layer consisted of nodes equal to the number of features. The first and second LSTM layer consisted of 20 and 10 nodes respectively, followed by another 10 nodes in the dense layer. The output layer consisted of 1 node, as is recommended for binary classification tasks. Other architectures was also considered, among others the architecture of Moen (2020). Still, in contrast to his findings, our architectural experimentation found an inclusion of two LSTM layers had a positive impact model performance. As discussed in Section 3.1.5, a notable issue of neural networks is its tendency to overfit. To counteract this a dropout layer of 0.1 was introduced. Visualizations of the two architectures, where the difference is the amount of input nodes, are presented in Figures 4.2 and 4.3

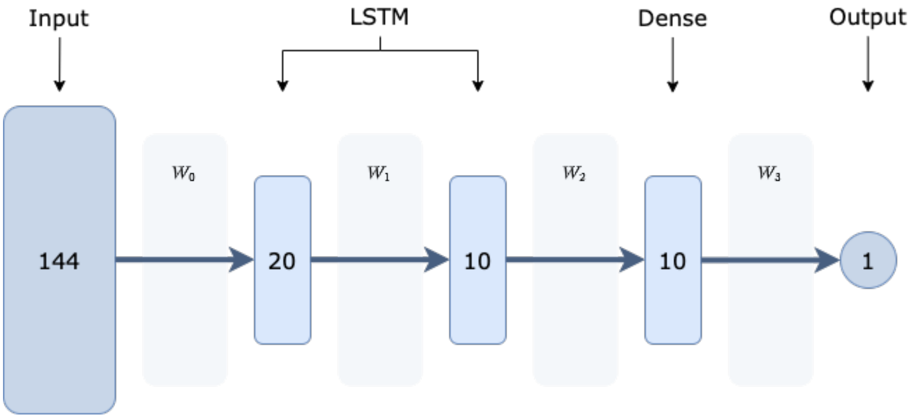


Figure 4.2: A visualization of the architecture of the neural network with 144 features. Note that the flatten and dropout layers are not represented

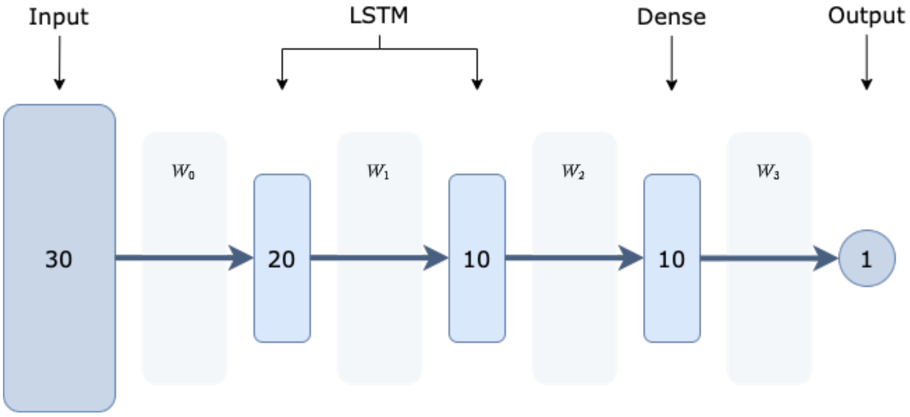


Figure 4.3: A visualization of the architecture of the neural networks with 30 features. Note that the flatten and dropout layers are not represented

The LSTM networks were trained using the binary cross-entropy loss function, despite the considerable drawbacks discussed in Section 3.1.2. ADAM (SGD) was chosen as the optimizer algorithm, with a learning rate of 0.00025 and a batch size of 32. This is the same setup as Moen (2020), which we also found during testing to be reasonable values.

The activation function for all hidden layers was set to tanh as it was the recommended sigmoidian function for hidden layers by Buttou et al. (2012). Still, Zhang et al. (2021) mentions that ReLU is another popular option, and was therefore also tested. In the output layer, the sigmoid activation function was used to enable the output to be interpreted as probabilities of bankruptcy.

The model containing all features used for feature selection, was trained for 30 epochs, as it was deemed sufficient for model training, while not overfitting on the training data. Meanwhile for the models containing the 30 selected features, 20 epochs was used as overfitting happened earlier. Alternative ways of reducing overfitting was also considered in addition to the dropout layer, mainly an early stopping condition. However, as this condition is based on the validation loss no longer improving (Chollet, 2018), and since the chosen loss function did not fully correlate to the success of the task (Section 3.1.2) consequently resulting in rather erratic validation losses, this method deemed unfit for our purpose.

4.4.2 Baseline neural networks implementation

In order to better evaluate the performance of the LSTM networks, we further developed two additonal deep neural networks as baseline models. The first is a traditional RNN trained on the same data as the LSTM network utilizing a sequence of four accounting years and the feature subset. In the Keras library a traditional RNN layer is called a “SimpleRNN” layer. The second is a deep fully connected feed-forward network, not capable of utilizing sequential data. In the Keras Python package, a fully connected layer is called a “dense” layer. When naming the neural neural networks (Section 4.4.3) we use the terminology from Keras, and refer to the traditional RNN as SimpleRNN and the fully connected feed-forward network as densely connected network. Both baseline networks have the same basic architecture as the LSTM networks, but with slight differences. For the SimpleRNN we switched the two LSTM layers for SimpleRNN layers. Likewise, for the densely connected network the two LSTM layers was changed to dense layers, and the flatten layer removed. All other parameters was kept equal to the LSTM networks using the feature subset. The SimpleRNN used the training validation and test sets with four sequential years. The densely connected network used the sets without sequential data.

4.4.3 The naming of the networks

When naming the six networks created in this thesis, we wanted the names to clearly illustrate the differences between the networks for better readability. These differences come from three aspects: the network type, the sequence length and the amount of features used by the network. The networks are therefore named in the following way: $\text{Type}_{\text{Seq\_Feat}}$, where Type denotes the network type (i.e., LSTM, SimpleRNN or Dense), Seq denotes the sequence length (between 1 and 4), and Feat denotes the number of features (either all or 30). The names of all networks together with a description are presented in Table 4.4.

Table 4.4: Network names

Name	Description
LSTM$_{4\_all}$	The LSTM network using a sequence of four accounting years and all features.
LSTM$_{4\_30}$	The LSTM network using a sequence of four accounting years and the feature subset.
LSTM$_{3\_30}$	The LSTM network using a sequence of three accounting years and the feature subset.
LSTM$_{2\_30}$	The LSTM network using a sequence of two accounting years and the feature subset.
SimpleRNN$_{4\_30}$	The traditional RNN using a sequence of four accounting years and the feature subset.
Dense$_{1\_30}$	The densely connected network using the feature subset. Note that this type of network is not capable of using sequences of accounting data

4.5 Model evaluations

As mentioned in Section 2.3 our dataset is affected by severe imbalance, making common evaluation metrics such as accuracy inappropriate to use. To overcome this issue, AUC were utilized seen as an immune metric to class imbalance (Fawcett, 2006) and frequently used in research of bankruptcy prediction(Veganzones & Séverin, 2018). Jones (2017) notes that one of the benefits of the ROC curve is its visualization ability, making it easier for practitioners to determine the cutoff threshold balancing the sensitivity and specificity mentioned in Section 3.3.1, to manage the credit risk in accordance to the bank's risk and credit policy. To get an overview over the two types of errors can be of great value for practitioners since classifying a failing company as healthy is significantly more costly, than predicting a healthy company as failing (du Jardin, 2015; Lohmann & Ohliger, 2019; Stein, 2005). Still, note that no threshold value is set in this thesis.

As an addition to AUC, we made use of the Brier score. This is particularly useful when comparing models with almost identical value of the other evaluation metrics, as the Brier score tells how confident the model is in making its probability estimates as discussed in 3.3.2. Since some our models performed relatively the same with respect to the AUC score, it was appropriate to include Brier scores.

4.6 SHAP implementation

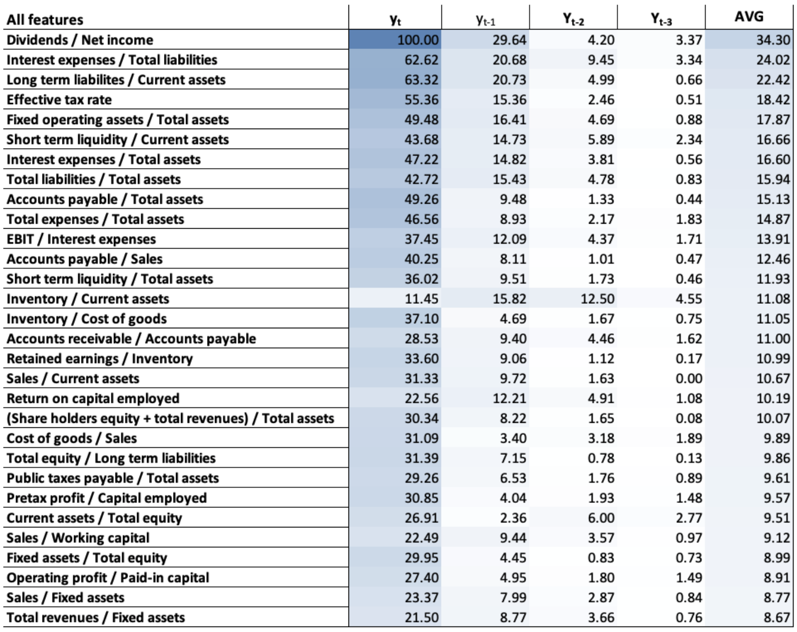
The implementation of the SHAP framework was done through the SHAP Python package (Lundberg, 2018b). Deep SHAP was used to calculate the approximate SHAP values for all models. To reduce computing time, we use a sample of 1000 observations from the training set. From there we can evaluate general feature effects and global feature importance through the SHAP summary plot and SHAP bar plot. For the local interpretations we generate SHAP waterfall plots, illustrating both effect and importance for each feature for a individual prediction. Note that other visualization methods are available through the SHAP Python package, such as force plot and decision plot. However, when using 30 features, we found the waterfall plot to be the best and most informative option.

It is important to note that as the models utilize sequential data, we calculate SHAP values for each features for all time-steps, consequently generating a way of observing the importance and effect of a feature over time. However, the global ranking based on impact magnitude of the features are based on the average across time steps. Further, no SHAP analysis of the SimpleRNN$_{4\_30}$ nor Dense$_{1\_30}$ were implemented, as the models were created as baseline models to evaluate the predictive performance of the LSTM networks compared to other deep neural networks for bankruptcy prediction.

Chapter 5

Results

In this chapter, the results of the analysis based on the methodology in chapter 4 are presented. Firstly, the performance of the neural networks based on the evaluation metrics are presented. This is followed by the results regarding SHAP. The discussion of the results is found in Chapter 6. For feature selection we chose a subset of 30 features based on the average magnitude of impact on model prediction from the SHAP analysis of LSTM$_{4\_all}$. The top 30 features are depicted in Figure 5.1.



All features	y_t	y_{t-1}	Y_{t-2}	Y_{t-3}	AVG
Dividends / Net income	100.00	29.64	4.20	3.37	34.30
Interest expenses / Total liabilities	62.62	20.68	9.45	3.34	24.02
Long term liabilites / Current assets	63.32	20.73	4.99	0.66	22.42
Effective tax rate	55.36	15.36	2.46	0.51	18.42
Fixed operating assets / Total assets	49.48	16.41	4.69	0.88	17.87
Short term liquidity / Current assets	43.68	14.73	5.89	2.34	16.66
Interest expenses / Total assets	47.22	14.82	3.81	0.56	16.60
Total liabilities / Total assets	42.72	15.43	4.78	0.83	15.94
Accounts payable / Total assets	49.26	9.48	1.33	0.44	15.13
Total expenses / Total assets	46.56	8.93	2.17	1.83	14.87
EBIT / Interest expenses	37.45	12.09	4.37	1.71	13.91
Accounts payable / Sales	40.25	8.11	1.01	0.47	12.46
Short term liquidity / Total assets	36.02	9.51	1.73	0.46	11.93
Inventory / Current assets	11.45	15.82	12.50	4.55	11.08
Inventory / Cost of goods	37.10	4.69	1.67	0.75	11.05
Accounts receivable / Accounts payable	28.53	9.40	4.46	1.62	11.00
Retained earnings / Inventory	33.60	9.06	1.12	0.17	10.99
Sales / Current assets	31.33	9.72	1.63	0.00	10.67
Return on capital employed	22.56	12.21	4.91	1.08	10.19
(Share holders equity + total revenues) / Total assets	30.34	8.22	1.65	0.08	10.07
Cost of goods / Sales	31.09	3.40	3.18	1.89	9.89
Total equity / Long term liabilities	31.39	7.15	0.78	0.13	9.86
Public taxes payable / Total assets	29.26	6.53	1.76	0.89	9.61
Pretax profit / Capital employed	30.85	4.04	1.93	1.48	9.57
Current assets / Total equity	26.91	2.36	6.00	2.77	9.51
Sales / Working capital	22.49	9.44	3.57	0.97	9.12
Fixed assets / Total equity	29.95	4.45	0.83	0.73	8.99
Operating profit / Paid-in capital	27.40	4.95	1.80	1.49	8.91
Sales / Fixed assets	23.37	7.99	2.87	0.84	8.77
Total revenues / Fixed assets	21.50	8.77	3.66	0.76	8.67

Figure 5.1: SHAP table for LSTM$_{4\_all}$. Note that all features are scaled between 0 and 100. The darker the color, the higher the feature magnitude of impact compared to the other features.

5.1 Evaluation metrics and predictive performance

In this section the AUC and Brier score for each model is presented to evaluate the predictive performance of our models. We further compare the performance of the LSTM networks utilizing different sequence lengths, before comparing their performance with the baseline neural networks SimpleRNN$_{4\_30}$ and Dense$_{1\_30}$.

Table 5.1: AUC and Brier score for each model

Model	AUC	Brier score
LSTM$_{4\_all}$	0.9288	0.0477
LSTM$_{4\_30}$	0.9118	0.1313
LSTM$_{3\_30}$	0.8963	0.1052
LSTM$_{2\_30}$	0.8893	0.1051
SimpleRNN$_{4\_30}$	0.8962	0.0959
Dense$_{1\_30}$	0.8799	0.1377

Comparison of LSTM networks

We start by comparing the performance of the LSTM networks. From Table 5.1 we see a trend that for each sequential year omitted in the LSTM networks, the AUC gets slightly lower. The LSTM$_{4\_all}$ network is overall the best performing with an AUC of 0.9288 and a Brier score of 0.0477. There is a marginal disimprovement of the AUC (1.83%) between the LSTM$_{4\_all}$ network and LSTM$_{4\_30}$, indicating that using the feature subset reduces model performance. Additionally, there is a significant disimprovement in Brier score with an increase of 175.26%, indicating that when introducing the feature subset, the network become less confident in it's prediction.

Comparing the LSTM networks using the feature subset and different sequence lengths allows us to learn more about how additional time steps impact the predictive power of the LSTM networks. From Table 5.1 we see that the LSTM$_{4\_30}$ network has the superior AUC score of 0.9118 compared to the other LSTM networks using the feature subset. Comparing LSTM$_{4\_30}$ and LSTM$_{3\_30}$ we see a decrease of 1.70% in predictive performance in terms of AUC. However, there is a reduction of 19.88% in Brier score, meaning the LSTM$_{3\_30}$ is more confident in its predictions. Further, when omitting an additional time step, the performance based on AUC is reduced by 0.78% (between LSTM$_{3\_30}$ and LSTM$_{2\_30}$). Still, we observe no significant changes in Brier score between the two models. Together, this indicate that when more time steps are available to the LSTM networks, the performance is enhanced. On the other hand, in the case of Brier score, when reducing time steps, the networks become more confident in their predictions. This may be a consequence of smaller training sets for the networks with longer sequences.

Comparison of the LSTM networks with the baseline neural networks

Next, we compare the performance of the LSTM networks with the baseline neural networks. We start by comparing the LSTM$_{4\_30}$ and the SimpleRNN$_{4\_30}$, as they have equal sequence lengths, and are therefore trained on the same data, making them the most comparable. As is evident from Table 5.1, the LSTM$_{4\_30}$ network outperform the SimpleRNN$_{4\_30}$ in respect to AUC. When replacing the two LSTM layers with simpler traditional RNN layers, the AUC is reduced by 1.71%. This imply that the more advanced structure of the LSTM cell captures the sequential accounting information better, leading to improved predictive performance compared to traditional RNN cells. However, we observe that the SimpleRNN$_{4\_30}$ is significantly more confident in its predictions with a lower Brier score.

Furthermore, we observe that the performance of the LSTM$_{3\_30}$ network and the SimpleRNN$_{4\_30}$ are comparable, both in terms of AUC and Brier score. However, the SimpleRNN$_{4\_30}$ perform better than LSTM$_{2\_30}$ for bankruptcy prediction. This indicates that the sequence length is more influential on RNN performance, than the increased complexity and long-term capabilities of the LSTM cell, at least when the sequence length is short.

All the LSTM networks have higher predictive performance than Dense$_{1\_30}$ in temrs of our metrics. Notably, the introduction of sequences of four accounting years in the LSTM$_{4\_\text{all}}$ and LSTM$_{4\_30}$ lead to an increase of AUC by 5.56% and 3.63% respectively compared to Dense$_{1\_30}$. This despite the amount of training data being significantly higher for the densely connected network. Moreover, the two LSTM networks are both more confident in their predictions. Still, the difference in Brier score between LSTM$_{4\_30}$ and Dense$_{1\_30}$ is not substantial, only decreasing by 4.65%. However, we see that the more accounting years omitted from the sequences of financial statements, the closer the performance of the LSTM networks are to the Dense$_{1\_30}$ network. Notably, the increase of AUC for the LSTM$_{2\_30}$ is only 1.07% compared to Dense$_{1\_30}$. These findings suggest, in line with our expectations, that networks capable of processing sequential accounting data are better and more appropriate for predicting bankruptcy than models neglecting the time dimension.

Comparison to previous work

We will now compare the predictive performance of our LSTM networks with previous literature in the domain. However, as stated in Section 3.1.5, neural networks are notoriously data hungry, meaning comparing the predictive performance with previous work using different data can be problematic. Still, as Moen (2020) used a previous version of the same dataset utilized in this thesis and approximately the same feature set, it is feasible to compare our findings. Moen (2020) achieved an out-of-sample AUC of 0.8836 and a Brier score of 0.1295 for his LSTM network using a sequence of four accounting years. Comparing this to our similar LSTM_{4_30} (same amount of features and sequence length) network we see that our model achieved a higher AUC score of 0.9118. meaning an increase of 3.19% compared to the LSTM network of Moen (2020). However, we also see that his model has a slightly lower Brier score compared to our network.

ROC curve

By looking at the ROC curve in Figure 5.2 we can visually see the distribution of the sensitivity (true positive rate) and 1− specificity (false positive rate) are rather close to each other for every model. However, the plot also depicts that the overall performance of the LSTM networks increases when including more time-steps and more features. Further, the plot illustrates that the performance of Dense_{1_30} is lower than the other models.

Probability distribution

Figure 5.3 depicts the prediction probability distribution for all models. It shows heavy tails for the lowest probabilities for the LSTM networks and the SimpleRNN_{4_30}, where the $\text{LSTM}_{4\_\text{all}}$ network have the heaviest tail in the lower intervals. This difference in distribution is natural when the model is so confident in its predictions (Brier score), and the fact that the actual amount of bankrupt companies is very low in the dataset. Still, for the other LSTM networks and the SimpleRNN_{4_30} the probability distributions are comparable. The plot also depicts that the predicted probability distribution for the Dense_{1_30} differs significantly from the other networks, not having the same clear tail in the lower intervals. The amount of predicted probabilities decline almost linearly with increased predicted probabilities. This may indicate that the RNN and LSTM networks adapt better to the true data

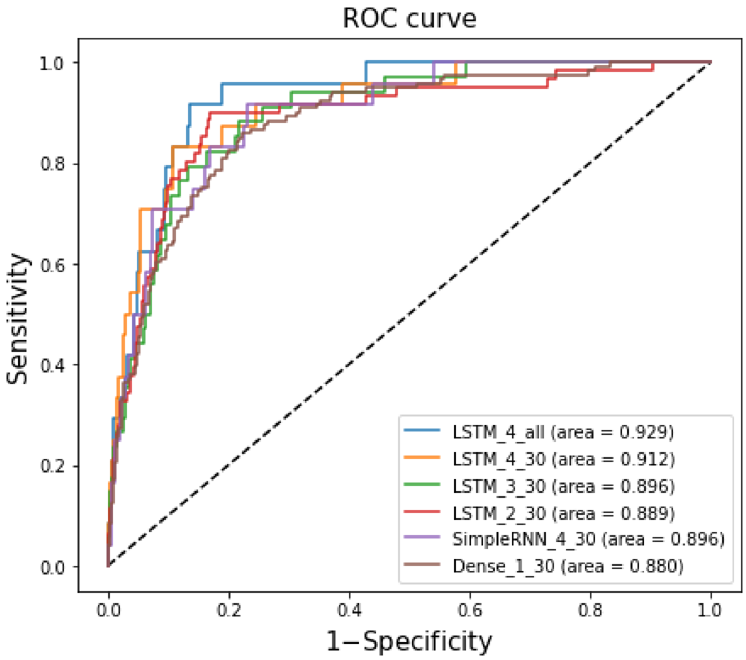


Figure 5.2: ROC curve and AUC for all models. The x-axis is $1-$Specificity (false positive rate), and the y-axis is sensitivity (true positive rate). The plot depicts that the overall performance of the LSTM networks increases when including more time-steps and more features. It also shows that the LSTM networks outperform the densely connected neural network.

distribution, and therefore achieve lower Brier scores. Still, this has not resulted in a large reductions in predictive performance in terms of AUC.

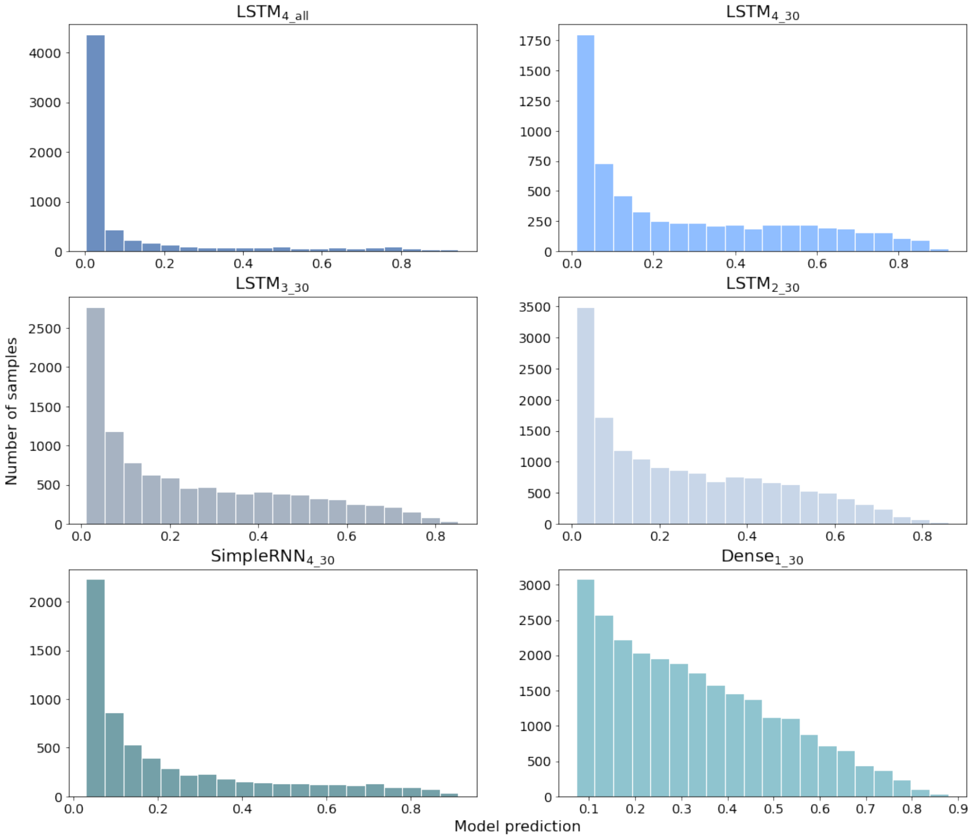


Figure 5.3: The prediction probability distribution for all models. It shows heavy tails on the lower intervals for the LSTM networks and the RNN.

Summary of predictive performance

To summarize we generally see that LSTM networks utilizing sequential accounting data perform better compared the baseline neural networks. Furthermore, we observe that for each time step omitted from the LSTM networks, the performance is reduced. Therefore, our findings indicate that LSTM networks with longer sequences of accounting data is to be preferred for bankruptcy prediction based solely on predictive performance. Still, for increased real world adoption of LSTM networks, we need to evaluate the model interpretability.

5.2 SHAP for global explanations

This section concerns the results from the SHAP evaluation for global explanations. First, we present the results regarding SHAP for impact magnitude, followed by SHAP for general feature effects. We have implemented the SHAP framework for increased interpretability of all LSTM networks, and will compare the findings of each LSTM network. Still, in this section, only the figures concerning the LSTM$_{4\_30}$ will be presented. The rest can be found in Appendix A, B and C. The implications of SHAP for increased model interpretability and consequently industry adoption is discussed in Section 6.4.

5.2.1 SHAP for global impact magnitude

Through equation 3.11 on page 36 we generate the average feature magnitude, and sort by feature importance, generating the SHAP bar plot in Figure 5.4. Note that as mentioned in Section 4.6 we generate the average SHAP magnitude for each time step, though the ordering is based on the average across time.

Figure 5.4 shows the average magnitude of SHAP values for all time steps in the LSTM$_{4\_30}$ network. The plot shows that `Dividends / Net income` is the most important feature for predicting bankruptcy. Additionally, `Total liabilities / Total assets` and `Effective tax rate` have high average impact on model prediction. Further, this plot clearly indicate that generally, the impact of a feature on the model prediction is reduced across time. Notably, features from more than two years prior to the last accounting year in the sequence have considerably less impact on model prediction in relation to the two latest years. For instance, `Dividends / Net income` from the last two years (Y_t and Y_{t-1}) have considerably greater impact on the model than Y_{t-2} and Y_{t-3}. This means that the LSTM$_{4\_30}$ network have learned that the dividends payed in relation to net income more than two years ago have severely less impact on whether the company goes bankrupt or not compared to the last two years. Intuitively, these findings make sense. Still, some features of Y_{t-1} have higher impact on model prediction than other features of Y_t. Notably, the SHAP analysis indicate that `Dividends / Net income` in Y_{t-1} have greater impact on the model prediction than for instance `Inventory / Current assets` in the latest accounting year (Y_t).

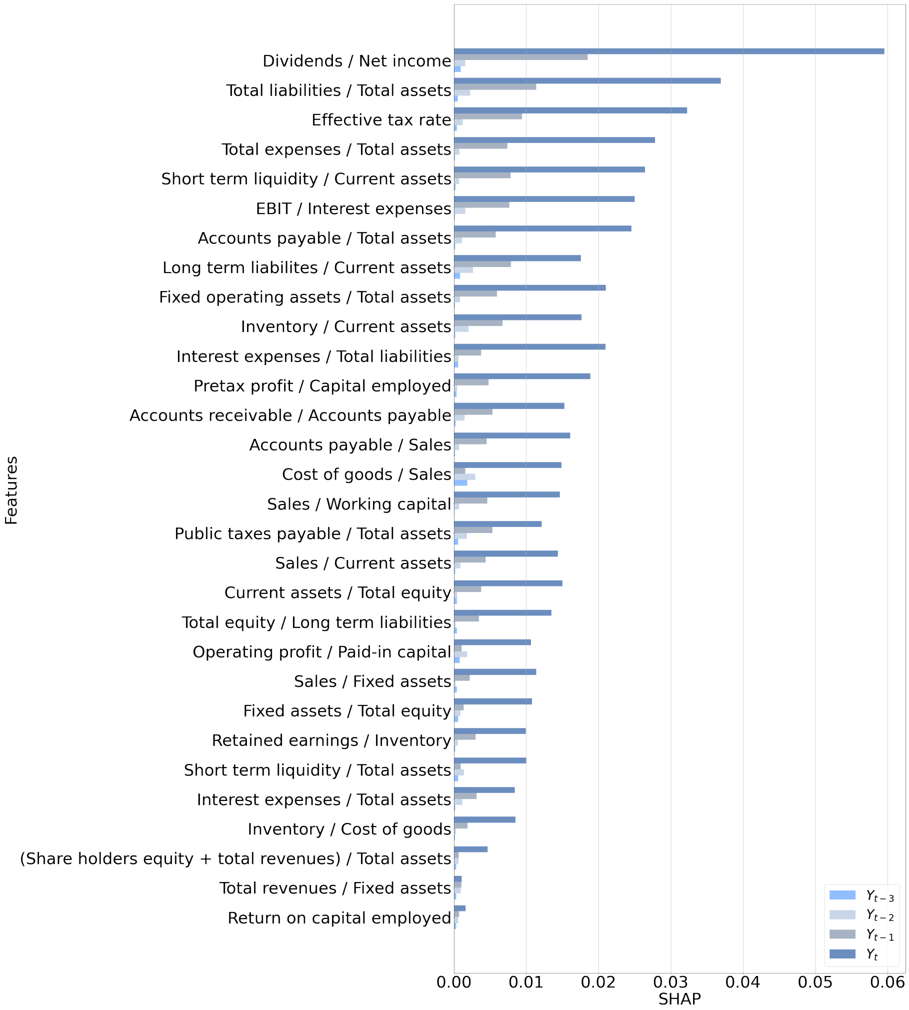
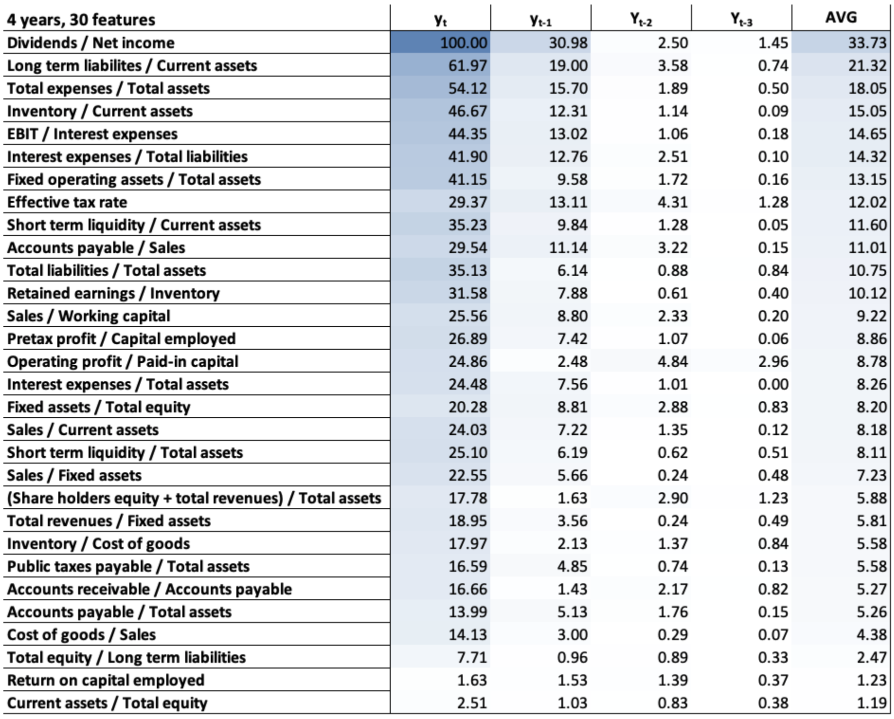


Figure 5.4: The SHAP bar plot for the LSTM$_{4\_30}$ network. We observe that the feature impact magnitude generally decreases for each time step, where the features of Y_t generally have the most impact on model prediction.

SHAP table for global feature impact magnitude

Figure 5.5 provides an alternative visualisation to the SHAP bar plot for the LSTM_{4_30} network, and also shows the magnitude of impact for each feature for each year. The average feature magnitude is also depicted. Note that the SHAP values are standardized between 0 and 100, and that a darker color indicate a higher SHAP value.



4 years, 30 features	Y_t	Y_{t-1}	Y_{t-2}	Y_{t-3}	AVG
Dividends / Net income	100.00	30.98	2.50	1.45	33.73
Long term liabilites / Current assets	61.97	19.00	3.58	0.74	21.32
Total expenses / Total assets	54.12	15.70	1.89	0.50	18.05
Inventory / Current assets	46.67	12.31	1.14	0.09	15.05
EBIT / Interest expenses	44.35	13.02	1.06	0.18	14.65
Interest expenses / Total liabilities	41.90	12.76	2.51	0.10	14.32
Fixed operating assets / Total assets	41.15	9.58	1.72	0.16	13.15
Effective tax rate	29.37	13.11	4.31	1.28	12.02
Short term liquidity / Current assets	35.23	9.84	1.28	0.05	11.60
Accounts payable / Sales	29.54	11.14	3.22	0.15	11.01
Total liabilities / Total assets	35.13	6.14	0.88	0.84	10.75
Retained earnings / Inventory	31.58	7.88	0.61	0.40	10.12
Sales / Working capital	25.56	8.80	2.33	0.20	9.22
Pretax profit / Capital employed	26.89	7.42	1.07	0.06	8.86
Operating profit / Paid-in capital	24.86	2.48	4.84	2.96	8.78
Interest expenses / Total assets	24.48	7.56	1.01	0.00	8.26
Fixed assets / Total equity	20.28	8.81	2.88	0.83	8.20
Sales / Current assets	24.03	7.22	1.35	0.12	8.18
Short term liquidity / Total assets	25.10	6.19	0.62	0.51	8.11
Sales / Fixed assets	22.55	5.66	0.24	0.48	7.23
(Share holders equity + total revenues) / Total assets	17.78	1.63	2.90	1.23	5.88
Total revenues / Fixed assets	18.95	3.56	0.24	0.49	5.81
Inventory / Cost of goods	17.97	2.13	1.37	0.84	5.58
Public taxes payable / Total assets	16.59	4.85	0.74	0.13	5.58
Accounts receivable / Accounts payable	16.66	1.43	2.17	0.82	5.27
Accounts payable / Total assets	13.99	5.13	1.76	0.15	5.26
Cost of goods / Sales	14.13	3.00	0.29	0.07	4.38
Total equity / Long term liabilities	7.71	0.96	0.89	0.33	2.47
Return on capital employed	1.63	1.53	1.39	0.37	1.23
Current assets / Total equity	2.51	1.03	0.83	0.38	1.19

Figure 5.5: Table visualizing SHAP magnitude of importance for LSTM_{4_30}. Note that the SHAP values are standardized between 0 and 100, and a darker color indicate higher SHAP value.

Figure 5.5 depict a general pattern of lower values for each subsequent time step. This pattern is consistent for all models. Still, we see that some of the variables would have been ranked differently if only the first time step (accounting year) was utilized. For instance, `Effective tax rate` have a higher average magnitude of impact than `Short term liquidity / Current assets` due to generally higher SHAP magnitudes for the later time steps, even though the magnitude for Y_t is lower. Further, we see that `Operating profit / Paid-in capital` for Y_{t-1} has lower impact on model prediction than for Y_{t-2} and Y_{t-3}. Therefore, we observe some discrepancies from the general pattern.

Comparison of global feature impact between the LSTM networks

Generally, we observe that feature ranking for the models utilizing the feature subset are comparable, whereas the three most important features are all common. Moreover, `Dividends / Net income` is the upmost feature for all LSTM networks. Further, `Return on capital employed` resides towards the bottom for all models. However, there are some differences. Notably `Short term liquidity / Total asset` resides higher for the models with less time steps, being the LSTM$_{3\_30}$ network and the LSTM$_{2\_30}$ network. This may indicate that when including longer sequences of accounting data, short term liquidity seem less influential on bankruptcy probability.

5.2.2 SHAP for global feature effects

A SHAP summary plot describes how each feature influence the model prediction, ordered by average absolute SHAP value and hence feature importance. Each dot represents a SHAP value for a feature and an observation of that specific feature. The color indicate the feature value, where blue means low and red means high. The x-axis measure the SHAP value, where a negative value reduces the bankruptcy probability, while a positive increases the predicted bankruptcy risk. The plot therefore provide a view of how each feature generally impact the model prediction, or in other words, the feature effect. Figure 5.6 is the SHAP summary plot for the LSTM$_{4\_30}$ network. Note that all interpretations of feature effects is in relation to the other companies in the dataset.

As mentioned in Section 5.2.1, according to the SHAP analysis of the LSTM$_{4\_30}$ network, `Dividends / Net income` is the most important feature for predicting bankruptcy. From Figure 5.6 we see that a high feature value is often accompanied by lower (negative) SHAP values. This indicate that a company paying a big percentage of their net income as dividends have a lower probability of bankruptcy, meanwhile a company paying smaller percentages of their net income as dividends have a higher probability of bankruptcy.

We observe that a high `Total liabilities / Total assets` (debt ratio) generally increases the predicted bankruptcy probability. This means that the LSTM$_{4\_30}$ network have found a general pattern between higher debt ratio and bankruptcy probability. Furthermore, we see that a company with low average interest rate (`Interest expenses / Total liabilities`) compared to the other companies in the dataset, generally have a lower bankruptcy probability. Oppositely, a high average interest rate increases the chance of bankruptcy according to the SHAP analysis. Interestingly, we also see that a higher `Effective tax rate` reduces the probability of bankruptcy. At first glance,

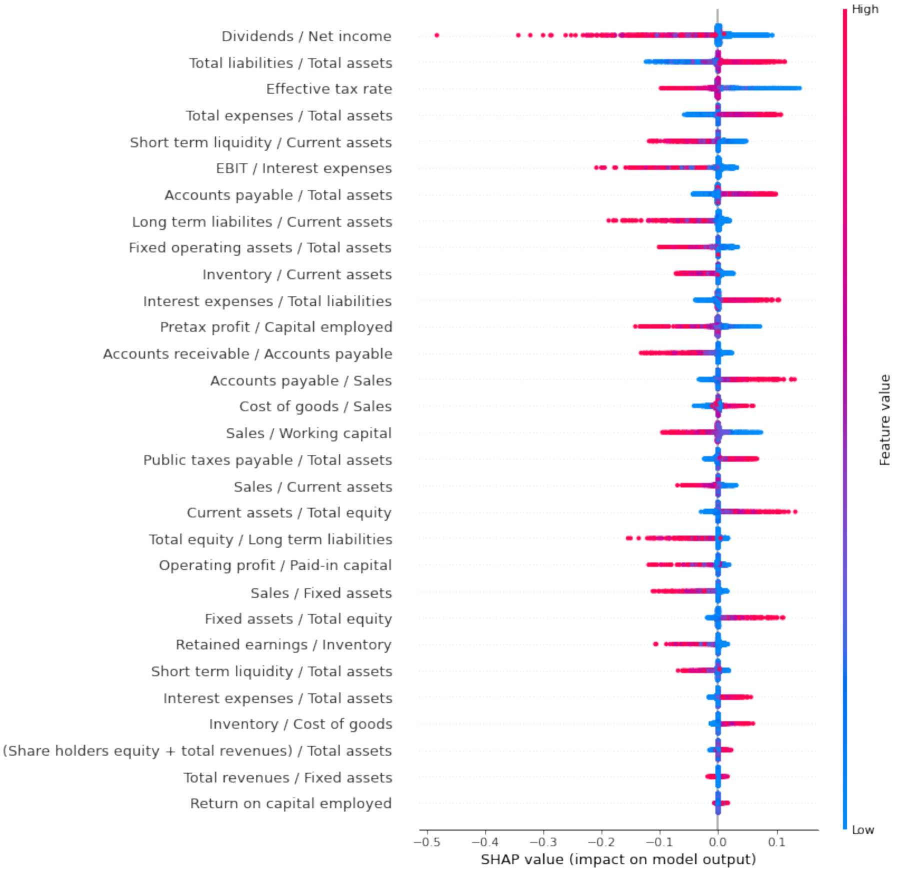


Figure 5.6: The SHAP summary plot for the LSTM$_{4\_30}$ network. The plot depicts the relationship between feature values and impact on model prediction. In other words, the general feature effect.

this relationship seem counter-intuitive. A discussion and our interpretation of this feature is found in Section 6.2.1.

Figure 5.6 also indicate that the model have found low feature values to generally have lower impact on model prediction, and is therefore often associated with lower SHAP value magnitude. Nevertheless, the plot illustrates that the model have generally found consistent feature effects across the board, were the plot show rather clear relationships between SHAP values and feature values for each individual variable. However, for `Total revenues / Fixed assets` this is not the case. We observe that the effect of a high feature value is associated with both increased and decreased bankruptcy probability, indicating that the model have not found a clear pattern of effect for the feature. For the other LSTM networks this feature have a more distinct effect, where a higher feature value is associated with reduced bankrupt risk.

Comparison of feature effects between LSTM networks

To increase trustworthiness of the deep LSTM networks, we compare the individual SHAP summary plots for each LSTM network to evaluate whether the learned behavior of each model is consistent. Generally the feature effects are consistent across the individual models. However, we observe some exceptions. Notably the general effect of `Sales / Current assets` for both the LSTM_{3_30} and the LSTM_{2_30} networks are opposite of the two other LSTM networks. Moreover, regarding the LSTM_{2_30} network, a similar observation can be seen for `Interest expenses / Total liabilities`. The SHAP analysis for the network indicated a relation were a high effective interest rate reduces bankruptcy probability, which is the opposite effect compared to the other LSTM networks.

5.3 SHAP for local explanations

So far, the results has concerned SHAP as a global explanation tool. Still, as explained in Section 3.2.2, SHAP is fundamentally a local explanation framework with the ability to give insight into the general logic of the model. The following results will therefore concern SHAP for local explanations to demonstrate how the framework enables interpretations of specific predictions for individual companies. We start by presenting a firm with a neutral predicted bankruptcy risk, before presenting two more companies with both high and low predicted bankruptcy probability from the LSTM_{4_30} network.

Neutral bankruptcy risk firm

Figure 5.7 depict the feature impact magnitude and effect for a specific prediction from the LSTM$_{4\_30}$ network. Note that the coloring of the waterfall plot is unrelated to the coloring of the summary plot. The colors strictly depict the sign of the SHAP values, where a positive SHAP value increases bankruptcy probability, while a negative have the opposite effect. The x-axis of the waterfall plot depicts the bankruptcy probability, while the y-axis depict the features ranked in descending order by magnitude of impact on the predicted probability of bankruptcy. By stacking the features on top of each other in this way, are we able to see the model's decision process for the specific prediction. We thereby interpret the plot by looking at both the feature impact magnitude, and the effect on the prediction for each feature. This enables a decision maker to understand what parts of their company are their greatest liabilities and strengths, and what decisions need to be made to reduce bankruptcy risk. A further discussion of the implications of local explanations for both decision makers, managers and financial institutions is found in Section 6.4.

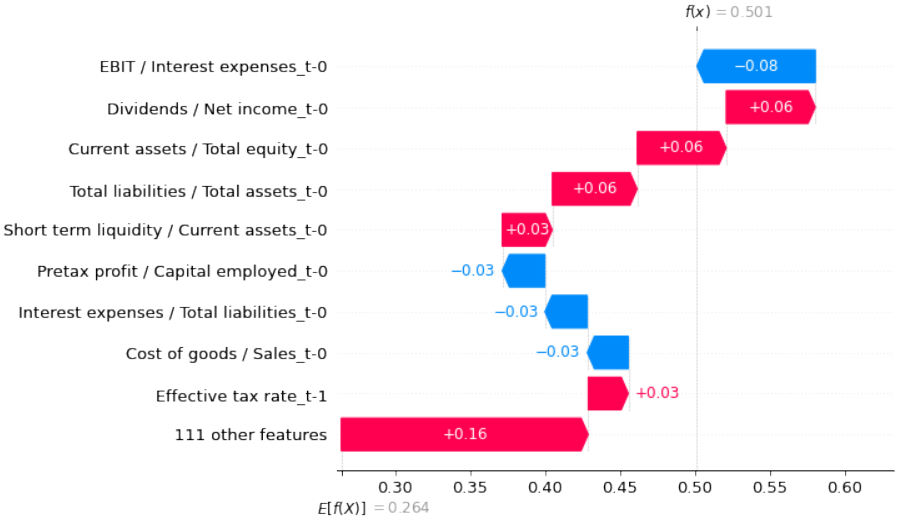


Figure 5.7: SHAP waterfall plot for illustrating feature contributions for the prediction of one individual observation from LSTM$_{4\_30}$ with a predicted bankruptcy probability of 0.501. Note that year $t - 0$ is equal to year t.

The predicted probability of bankruptcy, $f(x)$, for this instance is 0.501, were the expected/average probability $E[f(x)]$ is 0.264 and the sum of all feature effects is the difference in prediction of 0.237. This means that the LSTM$_{4\_30}$ network believes that there is an almost equal chance of the company going bankrupt or not. The feature with the greatest influence on this individual prediction is `EBIT / Interest expenses`$_{t-0}$ with an impact of -0.08, follow by `Dividends / Net income` $_{t-0}$. Interestingly, we see that `Effective tax rate`$_{t-1}$ have a greater impact on this specific prediction than the same feature for $t-0$. However, we also see that features from the last accounting year $t-0$ generally have the greatest impact magnitude. Lastly, the combined sum of all other features across all time steps not depicted in the figure are 0.16.

Table 5.2: Values and expected values for the features in Figure 5.7

Feature	Value	Expected value
EBIT/Interest expenses$_{t-0}$	0.2970	0.0945
Dividends/Net income$_{t-0}$	0	0.1605
Current assets/Total equity$_{t-0}$	0.7551	0.1889
Total liabilities/Total assets$_{t-0}$	0.8932	0.5795
Short term liquidity/Current assets$_{t-0}$	0	0.2814
Pretax profit/Capital employed$_{t-0}$	0.4883	0.3223
Interest expenses/Total liabilities$_{t-0}$	0.0015	0.2520
Cost of goods/Sales$_{t-0}$	0	0.3807
Effective tax rate$_{t-1}$	0	0.6019

Table 5.2 shows the scaled value of the features with the most impact for the individual prediction seen in Figure 5.7. By comparing the feature value to their expected value, are we able analyse whether or not the impacts for this observations are in line with the global explanations of feature effects presented in Section 5.2.2. We see that the value for the feature `EBIT / Interest expenses`$_{t-0}$ (interest coverage ratio) is much higher than the average, and influence the model prediction towards lower bankruptcy risk. This may indicate that the LSTM$_{4\_30}$ believes this specific company earns enough to cover its interest expenses, and therefore reduces the predicted bankruptcy probability based on this feature value. Our analysis of the SHAP summary plot in Figure 5.6 also indicated the same relationship. Therefore, this feature effect seem consistent with our global interpretations.

`Dividends / Net income`$_{t-0}$ has the opposite effect, and increases the predicted bankruptcy probability for this company. The value for this feature for this company is 0, and is therefore lower than the expected value of 0.1605. Compared to the general effects from Section 5.2.2, this relationship is also consistent. The same can be said for the rest of the features, all having the same effect as the general effect indicated from Figure 5.6.

High bankruptcy risk firm

To further illustrate the SHAP frameworks ability to give local explanations of an individual firm, we present a prediction from the LSTM$_{4\_30}$ network with a predicted probability of bankruptcy of 0.823. This means that the model predicts the firm to have high bankruptcy risk. From Figure 5.3 we observe that the most influential feature is `Accounts payable / Total assets`$_{t-0}$, directing the model to increase the predicted bankrupt probability of the firm. This is in line with the general feature effect in Figure 5.6.

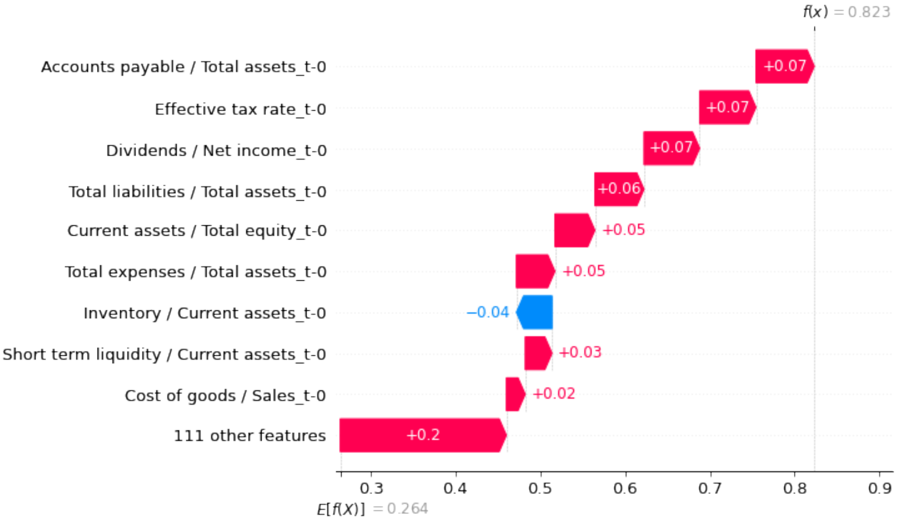


Figure 5.8: SHAP waterfall plot for illustrating feature contributions for the prediction of one individual observation from LSTM$_{4\_30}$ with a predicted bankruptcy probability of 0.823. Note that year $t - 0$ is equal to year t.

Moreover, we see that the firm has a high debt ratio (`Total liabilities / Total assets`$_{t-0}$) increasing the predicted bankruptcy probability similarly to the neutral risk firm presented previously in this section. We also observe that `Inventory / Current assets`$_{t-0}$ influences the model prediction towards lower bankruptcy probability as the feature value is high compared to other companies. Furthermore, Table 5.3 shows that the firm has a high `Cost of goods / Sales`$_{t-0}$, indicating a low contribution margin. The high value of this features increases the bankruptcy probability for this specific prediction, in line with the general logic of the model.

Table 5.3: Values and expected values for the features in Figure 5.8

Feature	Value	Expected value
Accounts payable/Total assets$_{t-0}$	1	0.2528
Effective tax rate$_{t-0}$	0.0486	0.5856
Dividends/Net income$_{t-0}$	0	0.1605
Total liabilities/Total assets$_{t-0}$	0.9149	0.5795
Current assets/Total equity$_{t-0}$	0.8481	0.1889
Total expenses/Total assets$_{t-0}$	0.7892	0.3515
Inventory/Current assets$_{t-0}$	1	0.2216
Short term liquidity/Current assets$_{t-0}$	0.00264	0.2814
Cost of good/Sales$_{t-0}$	0.9828	0.3807

Low bankruptcy risk firm

Lastly, we present an individual firm with a low risk of bankruptcy according the LSTM$_{4\_30}$ network. This company has a very low probability of bankruptcy of only 0.038. From Figure 5.9 we observe that the feature `Dividends / Net income`$_{t-0,t-1}$ have a substantial influence on the networks prediction, reducing the risk of bankruptcy with a greater magnitude of impact than the sum of all other features combined. Further, from Table 5.4 we can see that both these features have a value of 1, meaning the dividend payout to net income ratio have been very high for this company for the past two years compared to the other companies in the dataset. That `Dividends / Net income` have the greatest impact on this observation coincides with Figure 5.6 both in terms impact magnitude and feature effect. Further, the LSTM$_{4\_30}$ network found the features `Effective tax rate` and `Total liabilities / Total assets` for both $t-0$ and $t-1$ to be influential. In comparison, the high bankruptcy risk firm only found these features to have influence in $t-0$.

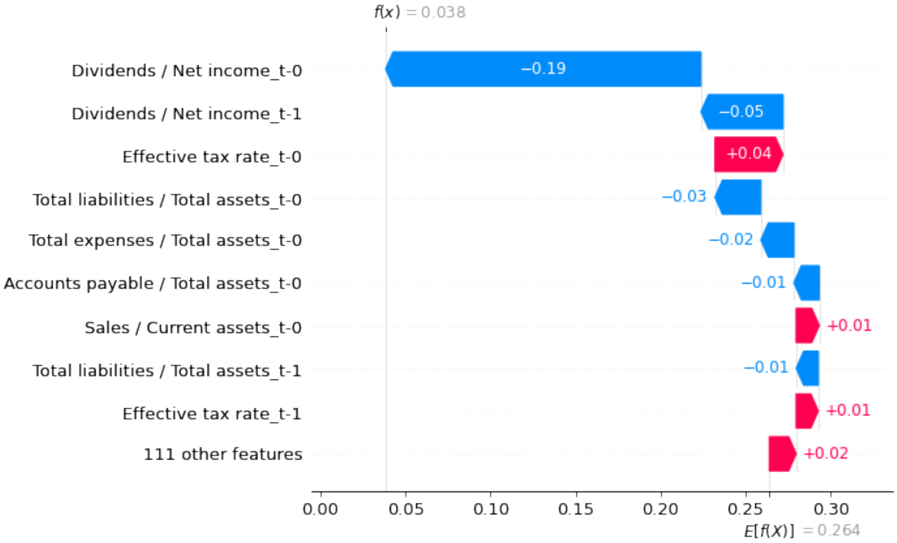


Figure 5.9: SHAP waterfall plot for illustrating feature contributions for the prediction of one individual observation from LSTM$_{4\_30}$ with a predicted bankruptcy probability of 0.038. Note that year $t - 0$ is equal to year t.

Table 5.4: Values and expected values for the features in Figure 5.9

Feature	Value	Expected value
Dividends/Net income$_{t-0}$	1	0.1605
Dividend/Net income$_{t-1}$	1	0.1548
Effective tax rate$_{t-0}$	0.0576	0.5856
Total liabilities/Total assets$_{t-0}$	0.2825	0.5795
Total expenses/Total assets$_{t-0}$	0.0050	0.3515
Accounts payable/Total assets$_{t-0}$	0.0019	0.2528
Sales/Current assets$_{t-0}$	0.2413	0.3545
Total liabilities/Total assets$_{t-1}$	0.0160	0.5851
Effective tax rate$_{t-1}$	0.0897	0.6019

Chapter 6

Discussion

In this chapter we discuss our findings, both regarding the predictive performance and the SHAP framework. We further present the limitations of the thesis, followed by a discussion concerning real-world implications of our findings.

6.1 Predictive performance

This section concerns the predictive performance of the LSTM networks. The findings presented in 5.1 suggest that there are differences in model performance between different sets of variables, time-steps and network types. These differences between the models can stem from multiple factors.

As presented in Section 2.3, feature selection have proven to increase model performance, by reducing the chance of the model disregarding minority samples as noise. However, in our case, the network using all features ($\mathrm{LSTM}_{4\_\mathrm{all}}$) both have higher predictive performance (AUC) and confidence (Brier score) compared to the LSTM_{4_30} network. This is in contradiction to the findings of Moen (2020). However, the difference between model AUCs is not massive, and partly confirm the findings of Paraschiv et al. (2021) that the model performance starts to flatten out after 25-30 features. Still, in our case, the feature selection did not increased model performance. One reason may be our use of SHAP for feature selection, even though the SHAP properties do not necessarily guarantee its suitability for such use. This means that we may not have found the optimal feature subset.

Nevertheless, the LSTM neural networks constructed in this thesis performs well according to the chosen metrics, even though the loss function does not fully correlate with the objective (Section 3.1.2). Despite the SHAP analysis indicating a strong reduction of feature importance across time steps for the LSTM networks, the inclusion of longer sequences of accounting data enables the models to perform better for bankruptcy prediction. This is also in spite that the amount of training data is reduced when the number of time steps is increased. However, using longer sequences of accounting data increase the Brier score after feature selection, meaning the LSTM networks lose some confidence in their predictions. A possible cause is that the amount of training data is increased when reducing sequence lengths. Together with the fact that neural networks are data-hungry, the increase of training data may reduce the variance in prediction, and making the model more assured. Still, there are no substantial changes in Brier score from including two or three years of accounting data, indicating that the reduction of time steps and increase of training data influenced the model by a similar magnitude.

Moreover, we see that the LSTM networks perform better than the two baseline deep neural networks being the SimpleRNN_{4_30} and the Dense_{1_30}. Firstly, a comparison of the SimpleRNN_{4_30} and the LSTM_{4_30} network shows a higher predictive performance for the LSTM network. This indicate that the addition of a cell state in the more complex LSTM cell enables the model to take better advantage of the four-year sequences of accounting data, in line with statements of Chollet (2018). Additionally, all LSTM networks outperform the Dense_{1_30} network, even though the amount of training data are significantly reduced. This is in line with the claims of Kim et al. (2020, 2021) who suggest that models utilizing sequential data are more suitable for bankruptcy prediction.

We achieved a high AUC for our LSTM_{4_30} network compared to the similar LSTM network of Moen (2020). The variation of AUC may stem from the differences in architecture of our respective models. Moen (2020) introduced only one LSTM layer into his network, whereas we found during testing that including two LSTM layers increased performance. This indicate that the more complex structure of our the LSTM networks enables the models to perform better for bankruptcy prediction, and that the addition of one more LSTM layer enables the network to better create representations for the input data. As stated in Section 5.1, compering the performance of our LSTM networks to other previous works within the domain can be problematic. Though Kim et al. (2021) only achieved and AUC of 0.68 for his LSTM network, and we, for our best performing network achieved an AUC of 0.9288, these results are not comparable. Most likely, our superior performance is not a consequence of a more optimized model, but rather the amount of training data, which is a very important aspect for complex neural networks. Nevertheless, we have proven that LSTM networks are indeed well suited for bankruptcy prediction. Additionally, we have shown that they outperform the baseline neural networks formed in our thesis by a small margin depending on sequence length, answering our first research question: "*To what extent can LSTM networks using sequential accounting data produce superior predictive performance compared to other neural network models for bankruptcy prediction?*".

6.2 SHAP explanations

In this section we first discuss the global explanations of SHAP, focusing on comparing the learned behaviour of LSTM_{4_30} with economic theory. Thereafter, we discuss our findings regarding SHAP for local explanations.

6.2.1 Global explanations

In order to utilize the SHAP framework to enhance applications of deep LSTM networks in a real world, the trustworthiness of the results need to be discussed. Therefore, we need to compare the general feature effects found by the models with economic theory, previous literature, and to some extent intuition. In doing so, we evaluate whether the model reasoning is sound, and consequently trustworthy. Note that not all features are discussed, only the features with the highest impact on bankruptcy prediction and other noteworthy findings. Additionally, as previously stated the SHAP analysis was done solely on the LSTM networks. Therefore, all statements in this sections only concern the LSTM networks.

The analysis in Section 5.2.1, indicated that the most important feature for predicting bankruptcy found by our models is `Dividends / Net income`, though the feature impact is significantly reduced for the accounting years three and four years prior to the latest financial statement. As stated in Section 2.5.3, Dielman and Oppenheimer (1984) claim that a company's dividends decisions could be of great importance in recognizing financial distress of the company. Further, Murekefu (2012) found a strong positive relationship between dividends payout and firm performance, indicating reduced risk of bankruptcy. This is also backed by Kanakriyah (2020). The relationship found by the model therefore seem trustworthy. Still, it should be noted, that this feature should partly be looked at as a profitability feature. Dividends is a distribution of profits to company shareholders. Therefore, if the company does not have any (or low) profits in the accounting year, more often than not, especially for SMEs, they will not pay any dividends. It is therefore feasible to assume that the LSTM networks have found this relationship, and that the feature indicate profitability of the company an addition to how the managers believe the financial situation to be.

Furthermore, `Total liabilities / Total assets` also have high SHAP magnitude for all models. Our LSTM networks have found that a higher debt ratio increases the probability of bankruptcy. This is supported by Salim and Yadav (2012) who found a higher debt ratio had a negative relationship with firm performance. Additionally Modina and Pietrovito (2014) found a negative relationship between debt ratio and financial distress. These findings seem intuitive, as a company with a high debt ratio are more vulnerable to volatile cash flows. Therefore, the learned behaviour of the networks concerning this feature seem trustworthy. Still, Ogachi et al. (2020) observed the opposite relationship between debt ratio and the risk of bankruptcy, meaning there are some inconsistencies regarding the effect of debt ratio on bankruptcy probability. Our models also found a relation between high effective interest rate and increased bankruptcy probability. However, the reverse relationship was not necessarily as clear, where our findings suggest that a low average interest rate does not necessarily indicate a lower bankruptcy risk. Still, it is reasonable to assume that it is preferable to have cheaper loans, rather than expensive ones. A high value of this feature may also indicate that a company needed to take on more short-term (and therefore expensive) loans to keep the business running, at least if the debt ratio is similar to other companies with a lower effective interest rate.

`Effective tax rate` is also a feature with considerable impact on model prediction for all LSTM networks. The networks have found that a high effective tax rate reduces the probability of bankruptcy. At first glance, this relationship seem a little odd. However, as companies with zero or negative profits all will have an effective tax rate of zero, this relationship with higher bankruptcy probability seem to hold true. Note that therefore this variable should be interpreted as a profitability variable, rather than a means of tax-increase argumentation. This relationship was also found by Moen (2020), for his logistic regression and CatBoost models. Still, this variable can therefore be rather misleading, and we argue that such variables are not necessarily fit for bankruptcy prediction models when intuitive interpretability is of importance.

Though we generally find the learned behaviour of the models to be consistent across all LSTM networks, and supported by economic theory and intuition, do some feature effects deviate from this. The LSTM_{3_30} and LSTM_{2_30} networks have the opposite feature effect for `Sales / Current assets` in relation to the two other LSTM networks. This learned behaviour is not necessarily consistent with economic theory, as it indicates that a higher value is associated with higher probability of bankruptcy, even though a higher value generally means that the business is generating revenue more efficiently. Furthermore, for the $\text{LSTM}_{4\_\text{all}}$ network, both a higher `Fixed` assets / Total equity and Current assets / Total equity results in higher bankruptcy probability. As these variables naturally have a negative correlation, this is inconsistent. However, for the LSTM networks using the feature subset, this relationship disappeared, meaning feature selection may increase consistency. Though these inconsistencies are rare, these findings may still, to some extent, reduce the trustworthiness of our LSTM networks.

Furthermore, some variables did not have clear effects. Notably, the learned behaviour regarding Inventory / Current assets for the $\text{LSTM}_{4\_\text{all}}$ network was inconsistent. A high value was associated with both lower and higher bankruptcy probabilities. The same can be said for Total revenues / Fixed assets in the LSTM_{4_30} network.

Additionally, we observe that generally high feature values often have greater impact on model prediction than lower values, being the case for both increased and decreased bankruptcy probability. A reason for this may be our assumption that all missing values is an observation of zero. Though this given correct accounting practices is true, when many companies do not have any value for the accounting item, the features utilizing said item automatically become zero. For instance, broadcasting companies do not necessarily have any inventory nor cost of goods, meaning a low value in Inventory / Cost of goods is natural. Therefore, as this feature do not relate to these types of companies, generating a value of zero, a model trained using this feature may not find a pattern between low values and bankruptcy probability. Another example is Dividends / Net income, the most influential feature for all LSTM networks. Many, especially SMEs, do not pay any dividends. This may be a reason for why the LSTM networks have found lower values of the feature to not have the same magnitude of impact on model prediction compared to a high value of the feature. Therefore, it may be more appropriate to choose more widely comparable features, or specializing models to predict bankruptcy in specific industries.

6.2.2 Local explanations

To analyse the SHAP frameworks local explanation abilities, we analysed three individual predictions from the LSTM_{4_30} network. Generally we observe that the feature effect in conjunction with the feature value is consistent between the individual predictions. Moreover, we see that some of the features with the greatest impact magnitude is the same across all three predictions, notably Dividends / Net income and Total liabilities / Total assets. However, some differences between the individual cases are also depicted. For instance, the most influential feature for the neutral bankruptcy risk firm depicted in Figure 5.7, is EBIT / Interest `expenses`$_{t-0}$, which is not represented in the top features for the other two companies. This illustrates that the LSTM$_{4\_30}$ network is able to formulate individual evaluations based on feature value for specific companies. Further, by comparing the specific feature effects combined with feature value of a individual predictions with the SHAP summary plot, can we evaluate whether the decision process is consistent with our understanding of the general logic of the the LSTM$_{4\_30}$ network. For all three individual predictions presented in this thesis, this seem to be the case, indicating trustworthiness of the networks predictions.

The insight into the decision process of the LSTM$_{4\_30}$ network for the neutral firm, can be leveraged by company decision makers to make better decisions going forward. To reduce the probability of bankruptcy, we see from Figure 5.7 combined with the Table 5.2, that the company should prioritize increasing their percentage of dividends payout (an explanation for this recommendation can be found in Section 6.2). Moreover, we see that the company has a high ratio of current assets to total equity, increasing the financial distress. The managers could therefore, to reduce the risk of bankruptcy, increase the company equity. This will also reduce the debt ratio, and further enhance the financial stability of the company, and reduce bankruptcy risk according to the SHAP analysis and the LSTM$_{4\_30}$ network.

For the high bankruptcy risk firm presented in Section 5.3, we observed that a high value of `Inventory / Current assets`$_{t-0}$ reduced predicted bankruptcy probability. This is interesting, as having a large inventory is usually expensive. Still, this may indicate that the LSTM$_{4\_30}$ network have found the company to have potential future income from sales. This feature effect is consistent with the general model logic, though it may not be consistent with economic theory, as assuming the model have found potential for future income by this variable may be a stretch to far. We also observe from Table 5.3 that the high risk firm has a high `Cost of goods / Sales`$_{t-0}$, indicating a low contribution margin. A low value of this features increases the bankruptcy probability and therefore seem intuitive. However, it should be noted that a low contribution margin does not necessarily indicate the products/business to be unprofitable. Some industries have naturally higher cost-of-goods, but low fixed expenses. Therefore, contribution margin should only be compared to companies in the same industry. This may also explain why there are some discrepancies in the SHAP summary plot in Figure 5.6. Still, we argue that we in this thesis have demonstrated the SHAP frameworks capabilities for local explanations, and therefore its usefulness as a decision making tool.

6.3 Limitations

In order to discuss the real-world applicability of deep neural networks for bankruptcy prediction based on the finding in this thesis, our limitations need to be highlighted.

Firstly, as presented in Section 2.3, Fryer et al. (2021) suggests that the SHAP framework is not necessarily fit for feature selection. Therefore, the features selection in this thesis may not accurately depict the true most influential features for bankruptcy prediction. Additionally, no corrections in regards to correlations between variables are built into the SHAP framework. Because of this, the impact magnitude of more general key figures such as liquidity or profitability may be spread across similar features. Consequently, even though the sum of all profitability features may result in high impact magnitude, this may not necessarily be represented by our feature subset, nor be intuitively depicted from the SHAP analysis. Therefore, even though all approximately fully correlated features were removed before feature selection, this is a major drawback, and can to some extent explain the differences in feature impact magnitude between the LSTM network containing all features and the models using the subset. This also taking into consideration the lower amount of features to distribute feature importance upon. Further, Balcaen and Ooghe (2006) argues that there are significant differences in the best predictor variables between data samples and countries. This further indicate that the feature impact magnitude and feature effects described in this thesis can not be generalized outside the population of Norwegian public SMEs. Nevertheless, the feature effects described by our SHAP analysis generally seem consistent with economic theory and previous literature.

To reduce the imbalanced dataset problem, a cost-sensitive learning strategy was implemented. However, as discussed in Section 2.3 the true cost of misclassification is hard, or even impossible to be sure of. Therefore, making assumptions regarding the cost of misclassification can be considered a limitation of the methodology of this thesis, as it may have introduced undetected bias into the networks.

No extensive individual hyperparameter tuning for each specific model were implemented. Consequently, as most of the hyperparameter testing happened in regards to the $\text{LSTM}_{4\_\text{all}}$ network, it is conceivable that the parameters better fit this model than the rest, resulting in reduced performance for the other LSTM networks. This can also explain why the performance decreased after feature selection, even though removing noise from the model is said to increase performance. This is also true for the two models constructed for comparison purposes.

The prediction horizon also needs to be considered. The inclusion of sequential data speaks to the possibility of prediction bankruptcy over multiple years. However, this thesis have predicted in a one year horizon, even though financial institutions and other stakeholders often seek the bankruptcy probability in the span of multiple years.

During data preprocessing we winzorized the features to protect against outliers while not removing observations. As all feature values above the 95th percentile was set to the percentile value, we end up with an abnormal amount of features with a value of 1 after standardization. This may reduce the trustworthiness of the SHAP analysis, especially in regards to local explanations and the interpretation of individual predictions in this thesis.

Because of the splitting scheme presented in 4.2, some considerations needs to be addressed regarding the time-leakage. Though the training and validation sets did not include the same companies, and thereby not the same data, they did contain the same accounting years. This meant that during training, the model was evaluated on the same time period as the training. Consequently, as we tried to find the correct amount of epochs to train the model based on the performance on the validation set, we may also have overfitted the networks for predicting bankruptcy for that specific period of time.

6.4 Real-world implications

The main reason for the lack of adoption of deep neural networks for bankruptcy prediction in practice is the lack of opacity and consequently interpretability and trustworthiness of deep neural networks. The goal of this thesis was therefore to train deep neural networks in a realistic setting using an imbalanced dataset and sequential data, while utilizing the SHAP framework to increase interpretability and consequently enhance real-world applicability. This part of the discussion will therefore examine if the SHAP framework enables real-world applications of deep recurrent neural networks for bankruptcy prediction. Therefore, this section is partly an extension of Section 6.2 combined with Section 6.3.

The specific criteria for adoption of deep neural networks in the bankruptcy prediction domain is not necessarily clear, except that they need to be accurate and interpretable. Additionally, it is reasonable to assume that financial institutions and other company stakeholders such as business leaders and employees have different focuses and preferences when it comes to model interpretability and predictive performance. However, the predictive performance of a model is undoubtedly important, as increased performance also enables better and more informed decision making. Consequently, adoption of high performance bankruptcy prediction tools, such as the LSTM networks created in this thesis, is something financial institutions and company stakeholders looking to use the model as a management tool also are interested in.

Using the SHAP framework, we have gained insight into the behaviour of the networks, being the main issue in regards to the moral hazards of opaque machine learning systems for practical use. This include the learned general logic, encompassing the feature importance across the time dimension, feature effects and magnitude of impact, and local explanations for individual predictions. Insight into the learned behaviour of the networks enables financial institutions to better understand the models and how the financial development of a firm affects the prediction. By analysing the feature effects, banks can gain insight on whether or not the output is trustworthy. From our analysis, the behaviour of the LSTM networks are generally consistent with economic theory and intuition. This greatly increases the trustworthiness of our deep LSTM neural networks for bankruptcy prediction through the SHAP analysis.

As for managers and decision makers, insight into a model's logic facilitates its use as a managerial tool. Mainly, it enables managers to illuminate issues in regards to their business, and make better and more informed decisions to reduce potential financial distress. This can be done by examining the impact of key features learned by the deep neural network, and further evaluate how these features influence bankruptcy probabilities. The managers can subsequently compare their own situation to the learned behaviour and correct their own deficiencies while ensuring efficient use of their companies scarce resources. As SHAP drastically increases interpretability of general model logic, it enables such considerations. Moreover, the local explanation capabilities of the SHAP framework enables managers and decision makers to understand what decisions need to be made specifically for their company.

The SHAP frameworks local explanations, are also preferred for companies applying for loans wanting to know why their application was denied. Moreover, a loan officer could utilize the local explanation to validate whether the model prediction is justified (Demajo et al., 2020). Furthermore, from a financial institution perspective, utilizing SHAP for individual explanations can with this in mind increase customer trust and loyalty. However, it should be noted that it is possible to create intentionally misleading interpretations of SHAP, as demonstrated by Slack et al. (2020). Therefore, financial institutions need to be transparent in their use of SHAP for model interpretations.

From the discussion in this section, we can evaluate whether the use of SHAP on deep LSTM networks provides sufficient answers to the questions about interpretability presented in Section 2.2.2. The conclusions are based on a comparison with the explainability of logistic regression models often used for bankruptcy prediction and known to be interpretable.

Our experiments suggest that the SHAP framework enables global explainability. Therefore, in regards to the question "*What drove the explanations more generally?*", we have demonstrated that SHAP enables such considerations. Moreover, we have demonstrated that the SHAP framework also facilitates local explanations, also answering the second question "*Which features with what effect mattered in individual predictions?*". Still, SHAP does not reduce model complexity, even though the interpretability is enhanced. Therefore, the third question "*How does the model work, and can it be easily explained?*", in terms of computations, algorithms and inner workings still remain challenging. However, as the SHAP framework increases both local and global interpretability, our analysis suggest that it facilitates adoption of deep LSTM neural networks for bankruptcy prediction in the domain, answering our second research question: "*How can the SHAP framework increase interpretability of deep recurrent neural networks for bankruptcy prediction, and to what extent can this facilitate the adoption of deep learning for bankruptcy prediction in the financial services sector?*".

Chapter 7

Conclusion

The objective of this thesis was to discuss and utilize the SHAP framework to increase model interpretability and enhance real-world applications of deep neural networks for bankruptcy prediction. To achieve this we constructed deep LSTM networks capable of using sequential accounting data to predict bankruptcy probabilities for Norwegian SMEs. A cost-sensitive learning strategy was implemented to handle the imbalanced dataset problem. Moreover, we analyzed the predictive performance of the LSTM networks using AUC and Brier score, comparing them to a traditional RNN and a fully connected feed-forward neural network to answer our first research question: "*To what extent can LSTM networks using sequential accounting data produce superior predictive performance compared to other neural network models for bankruptcy prediction?*". Further, we discuss whether the learned behaviour of the model appears trustworthy by comparing SHAP feature effects with economic theory and intuition. Lastly, we discuss the real-world implication of our analysis to answer our second research question: "*How can the SHAP framework increase interpretability of deep recurrent neural networks for bankruptcy prediction, and to what extent can this facilitate the adoption of deep learning for bankruptcy prediction in the financial services sector?*".

The out-of-sample performance of the deep LSTM networks were high compared to the fully connected feed-forward neural network. The LSTM network using a sequence of four accounting years and 144 features, obtained an AUC and a Brier score of 0.9288 and 0.0477 respectively. This was an increase of 5.56% in AUC and a decrease of 65.36% in Brier score compared to the fully connected feed-forward neural network. The performance of the LSTM networks steadily decreased when fewer time steps were available to the networks. This indicates that models with longer sequences of data indeed are better at predicting bankruptcy, and therefore may be preferable to other models strictly concerning predictive performance. Still, the analysis also identified that data from the last two accounting years in the sequence have higher influence on model prediction compared to the first two years. Moreover, when comparing our similarly structured LSTM network and traditional RNN, we see that the inclusion of LSTM cells increased the networks capabilities of remembering long-term dependencies, results in an increase of AUC by 1.74%. Though the performance increases were not substantial, our findings suggest that deep LSTM networks produce superior predictive performance compared to other neural networks for bankruptcy prediction.

Contrary to the standard practice in the domain of deep learning for bankruptcy prediction, a cost-sensitive learning strategy was implemented. We found the method to be a feasible alternative to resampling methods.

Lastly, we utilized and discussed the SHAP framework capabilities of increasing interpretability of the deep LSTM networks. We used SHAP for global explanations, and found the learned behavior of the model to be generally consistent with economic theory and intuition, increasing the trustworthiness of the LSTM networks predictions. We further evaluated the SHAP framework for local explanations. Our findings suggest that SHAP enables interpretations of both the features impact magnitude, and their effects on specific predictions from LSTM networks. However, SHAP do not reduce the complexity of LSTM networks. Therefore, explaining how LSTM networks work remain challenging. Still, on the basis that the SHAP framework enables both global and local explanations, the findings of this thesis suggest that the SHAP framework is a viable tool for reducing the black box problem, and facilitates adoption of deep recurrent neural networks for bankruptcy prediction in the financial services sector.

7.1 Future work

In section 6.3 we highlighted the limitations and considerations regarding our thesis. This section will consist of future work to address these issues in regards to bankruptcy prediction as a whole, and for applying deep learning methods for bankruptcy prediction in the real-world.

As mentioned, training neural network models for binary classification tasks when the class distribution is severely skewed is difficult. One reason for this is that the loss function usually maximizes classification rate (Section 2.3), and not other metrics such as AUC that is immune to class imbalance. Therefore, an application of the Mann-Whitney-Wilcoxon statistic as a loss function for neural networks for bankruptcy prediction could reduce the need to address the imbalanced dataset issue in other ways, consequently reducing the induced bias from these strategies and further promote real-world application.

In this thesis we only used a one-year prediction horizon, even though the use of sequential data speaks to the possibility of prediction bankruptcy over multiple years. Therefore, enlarging the predictive horizon is also something to be considered for future work, as company stakeholders are often interested in the bankruptcy probability over multiple periods. This also speaks to a continuation of utilizing sequential data for bankruptcy prediction.

In this thesis, we used data from companies across multiple industries. This means the LSTM networks tried to discover patterns between companies that are not necessarily comparable. For instance, contribution margins differ significantly between areas of business. Therefore, we argue that specializing deep neural networks for predictions within specific industries may be more appropriate. This will also enable the use of market specific variables, that may increase predictive performance.

There is a lack of studies regarding the criteria of bankruptcy prediction models for both financial institutions and company stakeholders. A qualitative analysis regrading their needs and wants could provide valuable insight into their specific requirements for adopting new bankruptcy prediction methods. This could provide guidelines for the continued development of explainable deep neural networks for bankruptcy prediction, and further facilitate its adoption in the real world.

References

Act relating to bankruptcy. (1984). Ministry of Justice; Public Security (LOV-1984-06-08-58).

Aggarwal, C. C. (2018). *Neural networks and deep learning: A textbook.* Springer International Publishing. http://link.springer.com/10.1007/978-3-319-94463-0

Agrawal, A., Gans, J., & Goldfarb, A. (2018). *Prediction machines: The simple economics of artificial intelligence.* Harvard Business Press.

Alaka, H. A., Oyedele, L. O., Owolabi, H. A., Kumar, V., Ajayi, S. O., Akinade, O. O., & Bilal, M. (2018). Systematic review of bankruptcy prediction models: Towards a framework for tool selection. *Expert Systems with Applications*, *94*, 164–184. https://doi.org/10.1016/j.eswa.2017.10.040

Alexandropoulos, S.-A., Aridas, C., Kotsiantis, S., & Vrahatis, M. (2019). A deep dense neural network for bankruptcy prediction. Springer Link.

Aljawazneh, H., Mora, A. M., García-Sánchez, P., & Castillo-Valdivieso, P. A. (2021). Comparing the performance of deep learning methods to predict companies' financial failure. *IEEE Access*, *9*, 97010–97038. https://doi.org/10.1109/ACCESS.2021.3093461

Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The Journal of Finance*, *23*(4), 589–609. https://doi.org/10.2307/2978933

Angelini, E., di Tollo, G., & Roli, A. (2008). A neural network approach for credit risk evaluation. *The Quarterly Review of Economics and Finance*, *48*(4), 733–755. https://doi.org/10.1016/j.qref.2007.04.001

Art. 22 GDPR – Automated individual decision-making, including profiling (2016). https://gdpr-info.eu/art-22-gdpr/

Athey, S. (2018). The impact of machine learning on economics. *The economics of artificial intelligence: An agenda* (pp. 507–547). University of Chicago Press. https://www.nber.org/books-and-chapters/economics-artificial-intelligence-agenda/impact-machine-learning-economics

Balcaen, S., & Ooghe, H. (2006). 35 years of studies on business failure: An overview of the classic statistical methodologies and their related problems. *The British Accounting Review*, *38*(1), 63–93. https://doi.org/10.1016/j.bar.2005.09.001

Bao, W., Lianju, N., & Yue, K. (2019). Integration of unsupervised and supervised machine learning algorithms for credit risk assessment. *Expert Systems with Applications*, *128*, 301–315. https://doi.org/10.1016/j.eswa.2019.02.033

Barnes, P. (1987). The analysis and use of financial ratios: A review article. *Journal of Business Finance & Accounting*, *14*(4), 449–461. https://doi.org/10.1111/j.1468-5957.1987.tb00106.x

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, *58*, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

Basel II: International convergence of capital measurement and capital standards: A revised framework. (2004). https://www.bis.org/publ/bcbs107.htm

Becerra-Vicario, R., Alaminos, D., Aranda, E., & Fernández-Gámez, M. A. (2020). Deep recurrent convolutional neural network for bankruptcy prediction: A case of the restaurant industry. *Sustainability*, *12*(12), 5180. https://doi.org/10.3390/su12125180

Bengio, Y., Courville, A., & Vincent, P. (2014, April 23). *Representation learning: A review and new perspectives* (arXiv:1206.5538) [type: article]. arXiv. http://arxiv.org/abs/1206.5538

Bernhardsen, E. (2007). Modellering av kredittrisiko i foretakssektoren - videreutvikling av SEBRA-modellen, 7.

Biran, O., & Cotton, C. (2017). Explanation and justification in machine learning: A survey. *IJCAI-17 workshop on explainable AI (XAI)*, *8*(1), 6.

Bracke, P., Datta, A., Jung, C., & Sen, S. (2019). Machine learning explainability in finance: An application to default risk analysis. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3435104

Brîndescu, D. (2016). Solvency ratio as a tool for bankruptcy prediction. *5*(2), 4.

Brynjolfsson, E., Hitt, L. M., & Kim, H. H. (2011). Strength in numbers: How does data-driven decisionmaking affect firm performance? *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.1819486

Burrell, J. (2016). How the machine ‘thinks’: Understanding opacity in machine learning algorithms. *Big Data & Society*, *3*(1), 2053951715622512. https://doi.org/10.1177/2053951715622512

Buttou, L., Orr, G. B., LeCun, Y. A., & Müller, K.-R. (2012). Efficient backprop. *Neural networks: Tricks of the trade* (pp. 9–48).

Campbell, J. Y., Hilscher, J., & Szilagyi, J. (2008). In search of distress risk. *The Journal of Finance*, *63*(6), 2899–2939. https://doi.org/10.1111/j.1540-6261.2008.01416.x

Chava, S., & Jarrow, R. A. (2004). Bankruptcy prediction with industry effects, 34.

Chen, N., Ribeiro, B., Vieira, A. S., Duarte, J., & Neves, J. C. (2011). A genetic algorithm-based approach to cost-sensitive bankruptcy prediction. *Expert Systems with Applications*, *38*(10), 12939–12945. https://doi.org/10.1016/j.eswa.2011.04.090

Chollet, F. (2018). *Deep learning with python*. Manning Publications Co.

Chou, T.-N. (2019). An explainable hybrid model for bankruptcy prediction based on the decision tree and deep neural network. *2019 IEEE 2nd International Conference on Knowledge Innovation and Invention (ICKII)*, 122–125. https://doi.org/10.1109/ICKII46306.2019.9042639

Classification on imbalanced data [TensorFlow]. (2022). https://www.tensorflow.org/tutorials/structured_data/imbalanced_data

Commission, E., for Small, E. A., Enterprises, M.-s., Muller, P., Devnani, S., Ladher, R., Cannings, J., Murphy, E., Robin, N., Ramos Illán, S., Aranda, F., Gorgels, S., Priem, M., Smid, S., Unlu Bohn, N., Lefebvre, V., & Frizis, I. (2021). *Annual report on european smes 2020/2021 : Digitalisation of smes : Background document* (K. Hope, Ed.). Publications Office. https://doi.org/doi/10.2826/120209

Cultrera, L., & Brédart, X. (2016). Bankruptcy prediction: The case of belgian SMEs. *Review of Accounting and Finance*, *15*(1), 101–119. https://doi.org/10.1108/RAF-06-2014-0059

Daubie, M., & Meskens, N. (2002). Business failure prediction: A review and analysis of the literature. In C. Zopounidis (Ed.), *New trends in banking management* (pp. 71–86). Physica-Verlag HD. https://doi.org/10.1007/978-3-642-57478-8_5

D’Aveni, R. A. (1989). The aftermath of organizational decline: A longitudinal study of the strategic and managerial characteristics of declining firms. *The Academy of Management Journal*, *32*(3), 577–605. https://doi.org/10.2307/256435

Deakin, E. B. (1972). A discriminant analysis of predictors of business failure. *Journal of Accounting Research*, *10*(1), 167–179. https://doi.org/10.2307/2490225

Demajo, L. M., Vella, V., & Dingli, A. (2020). Explainable AI for interpretable credit scoring. *Computer Science & Information Technology (CS & IT)*, 185–203. https://doi.org/10.5121/csit.2020.101516

Dielman, T. E., & Oppenheimer, H. R. (1984). An examination of investor behavior during periods of large dividend changes. *The Journal of Financial and Quantitative Analysis*, *19*(2), 197–216. https://doi.org/10.2307/2330898

Došilović, F. K., Brčić, M., & Hlupić, N. (2018). Explainable artificial intelligence: A survey. *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 0210–0215. https://doi.org/10.23919/MIPRO.2018.8400040

Du, M., Liu, N., & Hu, X. (2019). Techniques for interpretable machine learning. *arXiv:1808.00033 [cs, stat]*. http://arxiv.org/abs/1808.00033

Duan, J.-C., Sun, J., & Wang, T. (2012). *Multiperiod corporate default prediction - a forward intensity approach | elsevier enhanced reader.*

du Jardin, P. (2015). Bankruptcy prediction using terminal failure processes. *European Journal of Operational Research*, *242*(1), 286–303. https://doi.org/10.1016/j.ejor.2014.09.059

Eklund, T., Larsen, K., & Bernhardsen, E. (2001). Modell for analyse av kredittrisiko i foretakssektoren. *109-116*. https://norges-bank.brage.unit.no/norges-bank-xmlui/handle/11250/2480734

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Fenlon, C., O'Grady, L., Doherty, M. L., & Dunnion, J. (2018). A discussion of calibration techniques for evaluating binary and categorical predictive models. *Preventive Veterinary Medicine*, *149*, 107–114. https://doi.org/10.1016/j.prevetmed.2017.11.018

Filipe, S. F., Grammatikos, T., & Michala, D. (2016). Forecasting distress in european SME portfolios. *Journal of Banking & Finance*, *64*, 112–135. https://doi.org/10.1016/j.jbankfin.2015.12.007

Fryer, D., Strümke, I., & Nguyen, H. (2021). Shapley values for feature selection: The good, the bad, and the axioms. *IEEE Access*, *9*, 144352–144360. https://doi.org/10.1109/ACCESS.2021.3119110

Gawehn, E., Hiss, J. A., & Schneider, G. (2016). Deep learning in drug discovery. *Molecular Informatics*, *35*(1), 3–14. https://doi.org/10.1002/minf.201501008

Ghatasheh, N., Faris, H., Abukhurma, R., Castillo, P. A., Al-Madi, N., Mora, A. M., Al-Zoubi, A. M., & Hassanat, A. (2020). Cost-sensitive ensemble methods for bankruptcy prediction in a highly imbalanced data distribution: A real case from the spanish market. *Progress in*

Artificial Intelligence, *9*(4), 361–375. https://doi.org/10.1007/s13748-020-00219-x

Gordini, N. (2014). A genetic algorithm approach for SMEs bankruptcy prediction: Empirical evidence from italy. *Expert Systems with Applications*, *41*(14), 6433–6445. https://doi.org/10.1016/j.eswa.2014.04.026

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, *187*, 27–48. https://doi.org/10.1016/j.neucom.2015.09.116

Gupta, J., & Chaudhry, S. (2019). Mind the tail, or risk to fail. *Journal of Business Research*, *99*, 167–185. https://doi.org/10.1016/j.jbusres.2019.02.037

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, *73*, 220–239. https://doi.org/10.1016/j.eswa.2016.12.035

Härdle, W., Lee, Y.-J., Schäfer, D., & Yeh, Y.-R. (2009). Variable selection and oversampling in the use of smooth support vector machines for predicting the default risk of companies. *Journal of Forecasting*, *28*(6), 512–534. https://doi.org/10.1002/for.1109

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning – data mining, inference, and prediction* (2nd ed.). Springer.

He, H., & Garcia, E. A. (2009). Learning from imbalanced data [Conference Name: IEEE Transactions on Knowledge and Data Engineering]. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263–1284. https://doi.org/10.1109/TKDE.2008.239

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735. https://doi.org/10.1162/neco.1997.9.8.1735

Hosaka, T. (2019). *Bankruptcy prediction using imaged financial ratios and convolutional neural networks | elsevier enhanced reader*.

Jang, Y., Jeong, I., & Cho, Y. K. (2021). Identifying impact of variables in deep learning models on bankruptcy prediction of construction contractors. *Engineering, Construction and Architectural Management*, *28*(10), 3282–3298. https://doi.org/10.1108/ECAM-06-2020-0386

Janizek, J. D., Celik, S., & Lee, S.-I. (2018). *Explainable machine learning prediction of synergistic drug combinations for precision cancer medicine* (preprint). Cancer Biology. https://doi.org/10.1101/331769

Jones, S. (2017). Corporate bankruptcy prediction: A high dimensional analysis. *Review of Accounting Studies*, *22*(3), 1366–1422. https://doi.org/10.1007/s11142-017-9407-1

Kanakriyah, R. (2020). Dividend policy and companies' financial performance. *The Journal of Asian Finance, Economics and Business*, *7*(10), 531–541. https://doi.org/10.13106/jafeb.2020.vol7.no10.531

Kim, H., Cho, H., & Ryu, D. (2020). Corporate default predictions using machine learning: Literature review. *Sustainability*, *12*(16), 6325. https://doi.org/10.3390/su12166325

Kim, H., Cho, H., & Ryu, D. (2021). Corporate bankruptcy prediction using machine learning methodologies with a focus on sequential data. *Computational Economics*. https://doi.org/10.1007/s10614-021-10126-5

Kim, M.-J., Kang, D.-K., & Kim, H. B. (2015). Geometric mean based boosting algorithm with over-sampling to resolve data imbalance problem for bankruptcy prediction. *Expert Systems with Applications*, *42*(3), 1074–1082. https://doi.org/10.1016/j.eswa.2014.08.025

Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization. *arXiv:1412.6980 [cs]*. http://arxiv.org/abs/1412.6980

Kirkos, E. (2015). Assessing methodologies for intelligent bankruptcy prediction. *Artificial Intelligence Review*, *43*(1), 83–123. https://doi.org/10.1007/s10462-012-9367-6

Kou, G., Xu, Y., Peng, Y., Shen, F., Chen, Y., Chang, K., & Kou, S. (2021). Bankruptcy prediction for SMEs using transactional data and two-stage multiobjective feature selection. *Decision Support Systems*, *140*, 113429. https://doi.org/10.1016/j.dss.2020.113429

Krawczyk, B., Woźniak, M., & Schaefer, G. (2014). Cost-sensitive decision tree ensembles for effective imbalanced classification. *Applied Soft Computing*, *14*, 554–562. https://doi.org/10.1016/j.asoc.2013.08.014

Laitinen, E. (1993). Financial predictors for different phases of the failure process. *Omega*, *21*(2), 215–228. https://doi.org/10.1016/0305-0483(93)90054-O

Laitinen, E. K. (2005). Survival analysis and financial distress prediction: Finnish evidence. *Review of Accounting and Finance*, *4*(4), 76–90. https://doi.org/10.1108/eb043438

Laitinen, E. K., Lukason, O., & Suvas, A. (2014). Behaviour of financial ratios in firm failure process: An international comparison. *International Journal of Finance and Accounting*, 11.

Laitinen, T., & Kankaanpaa, M. (1999). Comparative analysis of failure prediction methods: The finnish case. *European Accounting Review*, *8*(1), 67–92. https://doi.org/10.1080/096381899336159

Lane, W. R., Looney, S. W., & Wansley, J. W. (1986). An application of the cox proportional hazards model to bank failure. *Journal of*

Banking & Finance, *10*(4), 511–531. https://doi.org/10.1016/S0378-4266(86)80003-6

Le, T., Vo, M. T., Vo, B., Lee, M. Y., & Baik, S. W. (2019). A hybrid approach using oversampling technique and cost-sensitive learning for bankruptcy prediction. *Complexity*, *2019*, e8460934. https://doi.org/10.1155/2019/8460934

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Lohmann, C., & Ohliger, T. (2019). The total cost of misclassification in credit scoring: A comparison of generalized linear models and generalized additive models. *Journal of Forecasting*, *38*(5), 375–389. https://doi.org/10.1002/for.2545

López, V., del Río, S., Benítez, J. M., & Herrera, F. (2015). Cost-sensitive linguistic fuzzy rule based classification systems under the MapReduce framework for imbalanced big data. *Fuzzy Sets and Systems*, *258*, 5–38. https://doi.org/10.1016/j.fss.2014.01.015

Lundberg, S. (2018a). *Shap.DeepExplainer*. https://shap-lrjball.readthedocs.io/en/latest/generated/shap.DeepExplainer.html

Lundberg, S. (2018b). *Welcome to the SHAP documentation*. https://shap.readthedocs.io/en/latest/index.html

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, *30*. Retrieved February 24, 2022, from https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html

Mansi, S., Maxwell, W., & Zhang, A. (2010). Bankruptcy prediction models and the cost of debt. *Journal of Fixed Income*, *21*. https://doi.org/10.2139/ssrn.1622407

Marcílio, W. E., & Eler, D. M. (2020). From explanations to feature selection: Assessing SHAP values as feature selection mechanism. *2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, 340–347. https://doi.org/10.1109/SIBGRAPI51738.2020.00053

Mensah, Y. M. (1984). An examination of the stationarity of multivariate bankruptcy prediction models: A methodological study. *Journal of Accounting Research*, *22*(1), 380–395. https://doi.org/10.2307/2490719

Modina, M., & Pietrovito, F. (2014). A default prediction model for italian SMEs: The relevance of the capital structure. *Applied Financial Economics*, *24*(23), 1537–1554. https://doi.org/10.1080/09603107.2014.927566

Moen, P. A. (2020). Bankruptcy prediction for norwegian enterprises using interpretable machine learning models with a novel timeseries problem formulation. https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2778369

Molnar, C. (2022). *Interpretable machine learning: A guide for making black box models explainable* (2nd ed.). https://christophm.github.io/interpretable-ml-book

Murekefu, T. M. (2012). The relationship between dividend payout and firm preformance: A study of listed companies in kenya. *8*, 18.

Nadar, D. S., & Wadhwa, B. (2019). Theoretical review of the role of financial ratios. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3472673

Odom, M., & Sharda, R. (1990). A neural network model for bankruptcy prediction. *1990 IJCNN International Joint Conference on Neural Networks*, 163–168 vol.2. https://doi.org/10.1109/IJCNN.1990.137710

Ogachi, D., Ndege, R., Gaturu, P., & Zoltan, Z. (2020). Corporate bankruptcy prediction model, a special focus on listed companies in kenya. *Journal of Risk and Financial Management*, *13*(3), 47. https://doi.org/10.3390/jrfm13030047

Ohlson, J. A. (1980). Financial ratios and the probabilistic prediction of bankruptcy. *Journal of Accounting Research*, *18*(1), 109–131. https://doi.org/10.2307/2490395

O'Shea, K., & Nash, R. (2015, December 2). *An introduction to convolutional neural networks* (arXiv:1511.08458). arXiv. http://arxiv.org/abs/1511.08458

Oxborough, C., Cameron, E., Rao, A., Birchall, A., Townsend, A., & Westermann, C. (2018). Explainable ai: Driving business value through greater understanding. *Retrieved from PWC website: https://www. pwc. co. uk/audit-assurance/assets/explainable-ai. pdf*.

Paraschiv, F., Schmid, M., & Wahlstrøm, R. R. (2021). Bankruptcy prediction of privately held SMEs using feature selection methods. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3911490

Park, M. S., Son, H., Hyun, C., & Hwang, H. J. (2021). Explainability of machine learning models for bankruptcy prediction. *IEEE Access*, *9*, 124887–124899. https://doi.org/10.1109/ACCESS.2021.3110270

Park, S., & Yang, J.-S. (2022). Interpretable deep learning LSTM model for intelligent economic decision-making. *Knowledge-Based Systems*, *248*, 108907. https://doi.org/10.1016/j.knosys.2022.108907

Parsa, A. B., Movahedi, A., Taghipour, H., Derrible, S., & Mohammadian, A. ((2020). Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis. *Accident Analysis & Prevention*, *136*, 105405. https://doi.org/10.1016/j.aap.2019.105405

Pelja, I., & Wahlstrøm, R. R. (2021). Hvordan påvirker bedriftens størrelse predikering av konkurs? *Tidsskrift for Økonomi Og Ledelse*, *7*, 82–91.

Qu, Y., Quan, P., Lei, M., & Shi, Y. (2019). Review of bankruptcy prediction using machine learning and deep learning techniques. *Procedia Computer Science*, *162*, 895–899. https://doi.org/10.1016/j.procs.2019.12.065

Razavi, S. (2021). Deep learning, explained: Fundamentals, explainability, and bridgeability to process-based modelling. *Environmental Modelling & Software*, *144*, 105159. https://doi.org/10.1016/j.envsoft.2021.105159

Salim, M., & Yadav, R. (2012). Capital structure and firm performance: Evidence from malaysian listed companies. *Procedia - Social and Behavioral Sciences*, *65*, 156–166. https://doi.org/10.1016/j.sbspro.2012.11.105

Salmi, T., & Martikainen, T. (1994). A review of the theoretical and empirical basis of financial ratio analysis. *Liiketaloudellinen aikakauskirja*, *43*(4). https://research.aalto.fi/en/publications/a-review-of-the-theoretical-and-empirical-basis-of-financial-rati

Schalck, C., & Yankol-Schalck, M. (2021). Predicting french SME failures: New evidence from machine learning techniques [Publisher: Routledge _eprint: https://doi.org/10.1080/00036846.2021.1934389]. *Applied Economics*, *53*(51), 5948–5963. https://doi.org/10.1080/00036846.2021.1934389

Shi, Y., & Li, X. (2019a). A bibliometric study on intelligent techniques of bankruptcy prediction for corporate firms. *Heliyon*, *5*(12), e02997. https://doi.org/10.1016/j.heliyon.2019.e02997

Shi, Y., & Li, X. (2019b). An overview of bankruptcy prediction models for corporate firms: A systematic literature review. *Intangible Capital*, *15*(2), 114. https://doi.org/10.3926/ic.1354

Shumway, T. (2001). Forecasting bankruptcy more accurately: A simple hazard model. *The Journal of Business*, *74*(1), 101–124. https://doi.org/10.1086/209665

Slack, D., Hilgard, S., Jia, E., Singh, S., & Lakkaraju, H. (2020). Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 180–186. https://doi.org/10.1145/3375627.3375830

Smiti, S., & Soui, M. (2020). Bankruptcy prediction using deep learning approach based on borderline SMOTE. *Information Systems Frontiers*, *22*(5), 1067–1083. https://doi.org/10.1007/s10796-020-10031-6

Son, H., Hyun, C., Phan, D., & Hwang, H. J. (2019). Data analytic approach for bankruptcy prediction. *Expert Systems with Applications*, *138*, 112816. https://doi.org/10.1016/j.eswa.2019.07.033

Stein, R. M. (2005). The relationship between default prediction and lending profits: Integrating ROC analysis and loan pricing. *Journal of Banking & Finance*, *29*(5), 1213–1236. https://doi.org/10.1016/j.jbankfin.2004.04.008

Tang, Y., Ji, J., Zhu, Y., Gao, S., Tang, Z., & Todo, Y. (2019). A differential evolution-oriented pruning neural network model for bankruptcy prediction. *Complexity*, *2019*, 1–21. https://doi.org/10.1155/2019/8682124

Tian, S., & Yu, Y. (2017). Financial ratios and bankruptcy predictions: An international evidence. *International Review of Economics & Finance*, *51*, 510–526. https://doi.org/10.1016/j.iref.2017.07.025

Tian, S., Yu, Y., & Guo, H. (2015). Variable selection and corporate bankruptcy forecasts. *Journal of Banking & Finance*, *52*, 89–100. https://doi.org/10.1016/j.jbankfin.2014.12.003

Tobback, E., Bellotti, T., Moeyersoms, J., Stankova, M., & Martens, D. (2017). Bankruptcy prediction for SMEs using relational data. *Decision Support Systems*, *102*, 69–81. https://doi.org/10.1016/j.dss.2017.07.004

Trinkle, B. S., & Baldwin, A. A. (2007). Interpretable credit model development via artificial neural networks. *Intelligent Systems in Accounting, Finance and Management*, *15*(3), 123–147. https://doi.org/10.1002/isaf.289

Van Gestel, T., Baesens, B., Suykens, J., Espinoza, M., Baestaens, D.-E., Vanthienen, J., & De Moor, B. (2003). Bankruptcy prediction with least squares support vector machine classifiers. *2003 IEEE International Conference on Computational Intelligence for Financial Engineering, 2003. Proceedings.*, 1–8. https://doi.org/10.1109/CIFER.2003.1196234

Veganzones, D., & Severin, E. (2020). Corporate failure prediction models in the twenty-first century: A review. *European Business Review*, *33*(2), 204–226. https://doi.org/10.1108/EBR-12-2018-0209

Veganzones, D., & Séverin, E. (2018). An investigation of bankruptcy prediction in imbalanced datasets. *Decision Support Systems*, *112*, 111–124. https://doi.org/10.1016/j.dss.2018.06.011

Vo, M., Vo, B., Lee, M., & Baik, S. (2019). A hybrid approach using oversampling technique and cost-sensitive learning for bankruptcy prediction. *Complexity*, *2019*, 1–12. https://doi.org/10.1155/2019/8460934

Vochozka, M., Vrbka, J., & Suler, P. (2020). Bankruptcy or success? the effective prediction of a company's financial development using LSTM. *Sustainability*, *12*(18), 7529. https://doi.org/10.3390/su12187529 von Eschenbach, W. J. (2021). Transparency and the black box problem: Why we do not trust AI. *Philosophy & Technology*, *34*(4), 1607–1622. https://doi.org/10.1007/s13347-021-00477-0

Wahlstrøm, R. R. (2022). Financial statements of companies in norway. https://doi.org/10.48550/arXiv.2203.12842

Wang, B. X., & Japkowicz, N. (2010). Boosting support vector machines for imbalanced data sets. *Knowledge and Information Systems*, *25*(1), 1–20. https://doi.org/10.1007/s10115-009-0198-y

Xiaomao, X., Xudong, Z., & Yuanfang, W. (2019). A comparison of feature selection methodology for solving classification problems in finance. *Journal of Physics: Conference Series*, *1284*(1), 012026. https://doi.org/10.1088/1742-6596/1284/1/012026

Yan, L., Dodier, R., Mozer, M. C., & Wolniewicz, R. (2003). Optimizing classifier performance via an approximation to the wilcoxon-mann-whitney statistic, 8.

Yang, E., Zhang, H., Guo, X., Zang, Z., Liu, Z., & Liu, Y. (2022). A multivariate multi-step LSTM forecasting model for tuberculosis incidence with model explanation in liaoning province, china. *BMC Infectious Diseases*, *22*(1), 490. https://doi.org/10.1186/s12879-022-07462-8

Yijing, L., Haixiang, G., Xiao, L., Yanan, L., & Jinling, L. (2016). Adapted ensemble classification algorithm based on multiple classifier system and feature selection for classifying multi-class imbalanced data. *Knowledge-Based Systems*, *94*, 88–104. https://doi.org/10.1016/j.knosys.2015.11.013

Zerilli, J., Knott, A., Maclaurin, J., & Gavaghan, C. (2019). Transparency in algorithmic and human decision-making: Is there a double standard? *Philosophy & Technology*, *32*(4), 661–683. https://doi.org/10.1007/s13347-018-0330-6

Zhang, A., Lipton, Z. C., Li, M., & Smola, A. J. (2021). Dive into deep learning, 839.

Zhang, J., & Thomas, L. (2015). The effect of introducing economic variables into credit scorecards: An example from invoice discounting. *Journal of Risk Model Validation*, *9*, 57–78. https://doi.org/10.21314/JRMV.2015.134

Zhang, X., Chen, X., Yao, L., Ge, C., & Dong, M. (2019). Deep neural network hyperparameter optimization with orthogonal array tuning. In T. Gedeon, K. W. Wong, & M. Lee (Eds.), *Neural information processing* (pp. 287–295). Springer International Publishing. https://doi.org/10.1007/978-3-030-36808-1_31

Zhou, L. (2013). Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods. *Knowledge-Based Systems*, *41*, 16–25. https://doi.org/10.1016/j.knosys.2012.12.007

Zmijewski, M. E. (1984). Methodological issues related to the estimation of financial distress prediction models [Publisher: [Accounting Research Center, Booth School of Business, University of Chicago, Wiley]]. *Journal of Accounting Research*, *22*, 59–82. https://doi.org/10.2307/2490859

Appendix A

SHAP summary plots

The SHAP summary plots for the LSTM$_{4\_all}$ network, the LSTM$_{3\_30}$ network and the LSTM$_{2\_30}$.

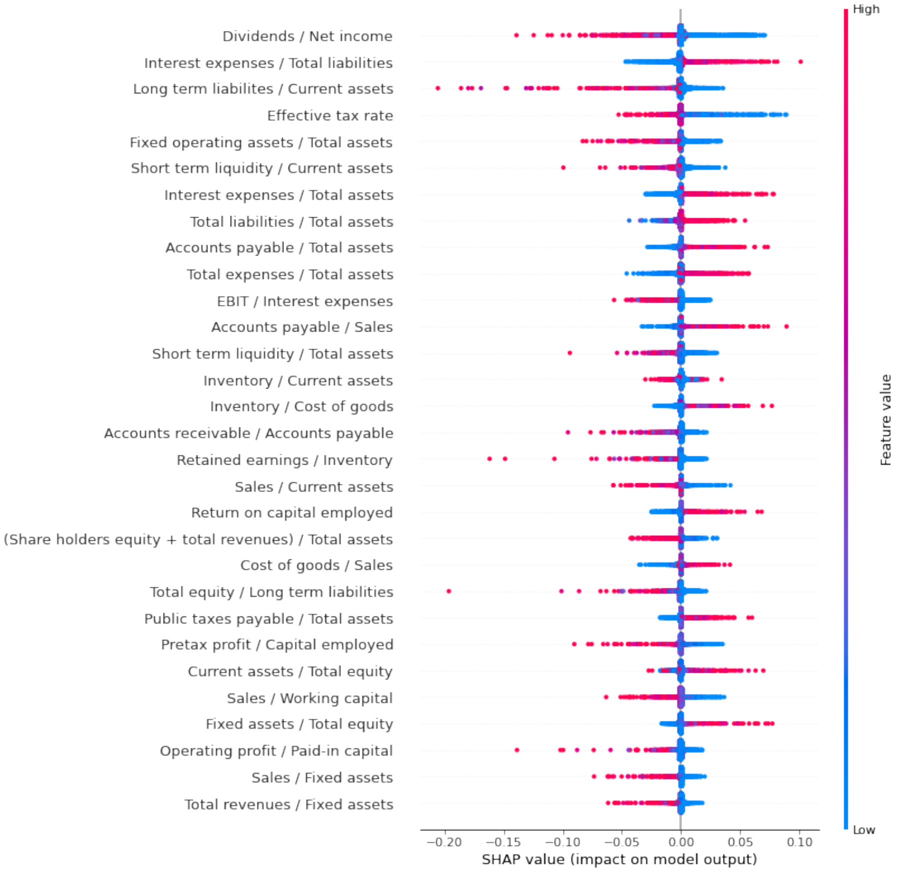


Figure A.1: The SHAP summary plot for LSTM$_{4\_all}$

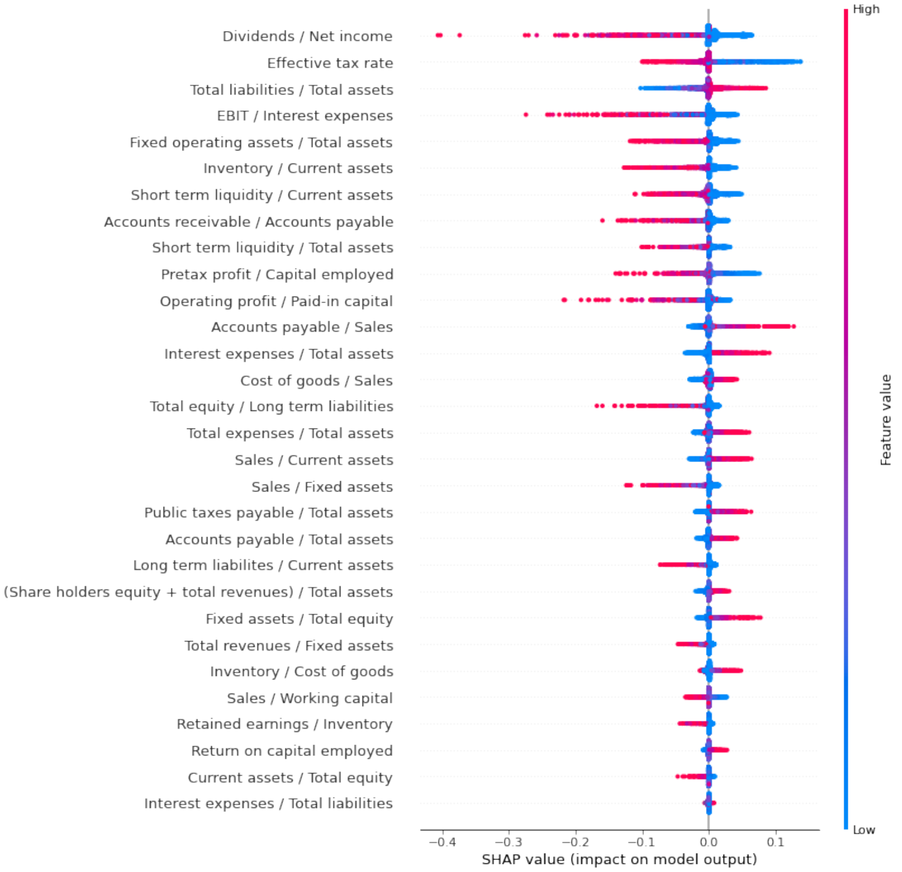


Figure A.2: The SHAP summary plot for LSTM$_{3\_30}$

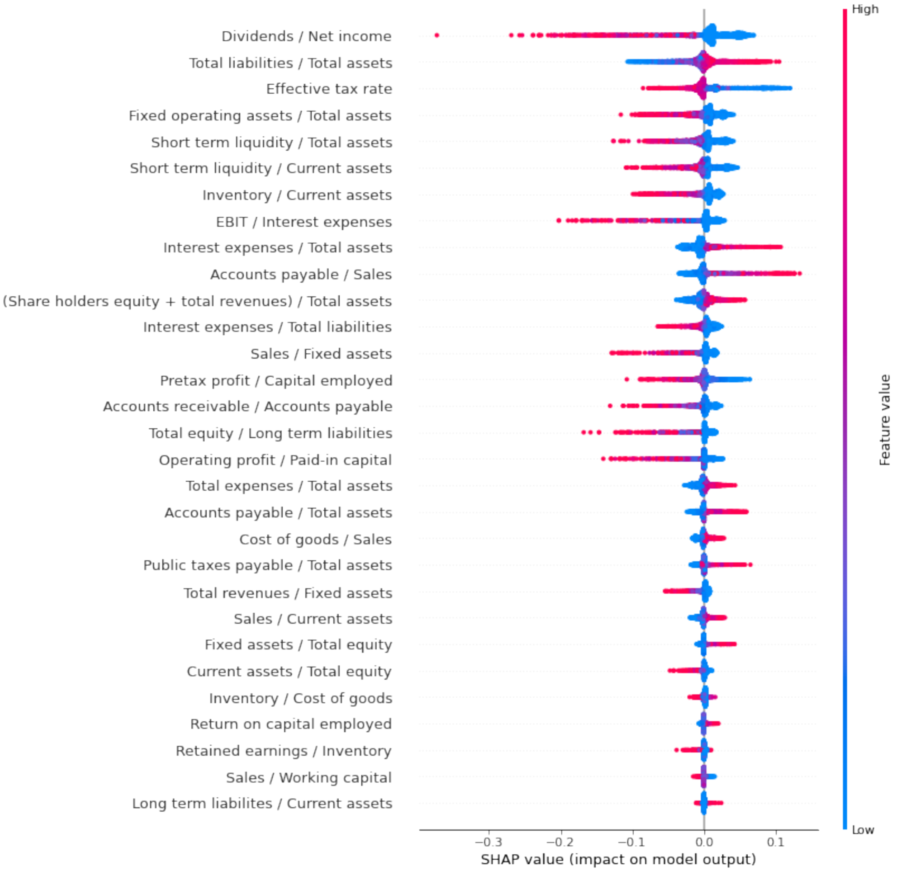


Figure A.3: The SHAP summary plot for LSTM_{2_30}

Appendix B

SHAP Bar plots

The SHAP bar plots for the LSTM$_{4\_all}$ network, the LSTM$_{3\_30}$ network and the LSTM$_{2\_30}$.

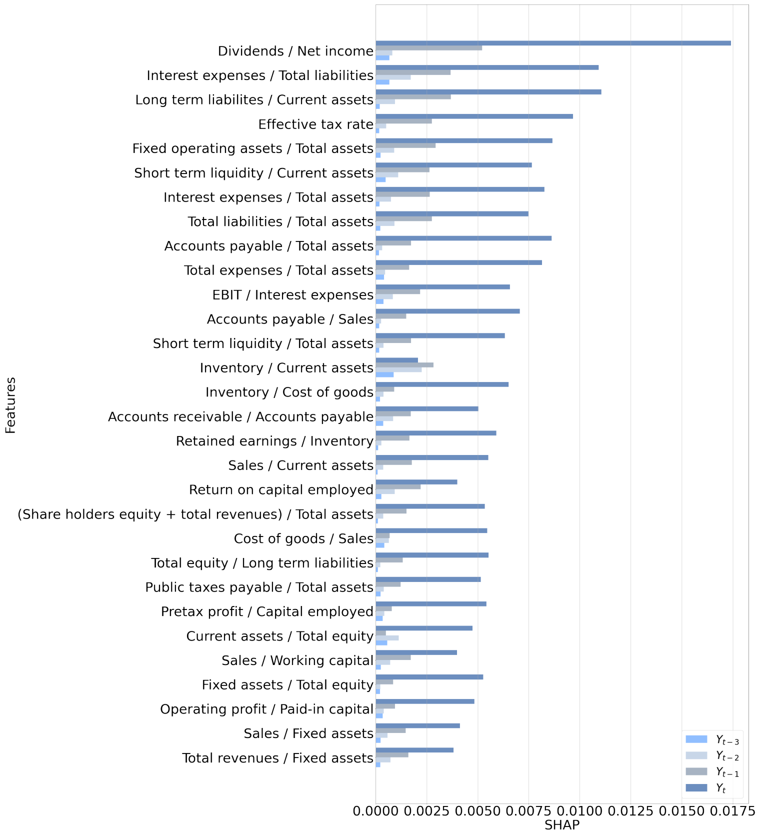


Figure B.1: The SHAP bar plot for LSTM$_{4\_all}$

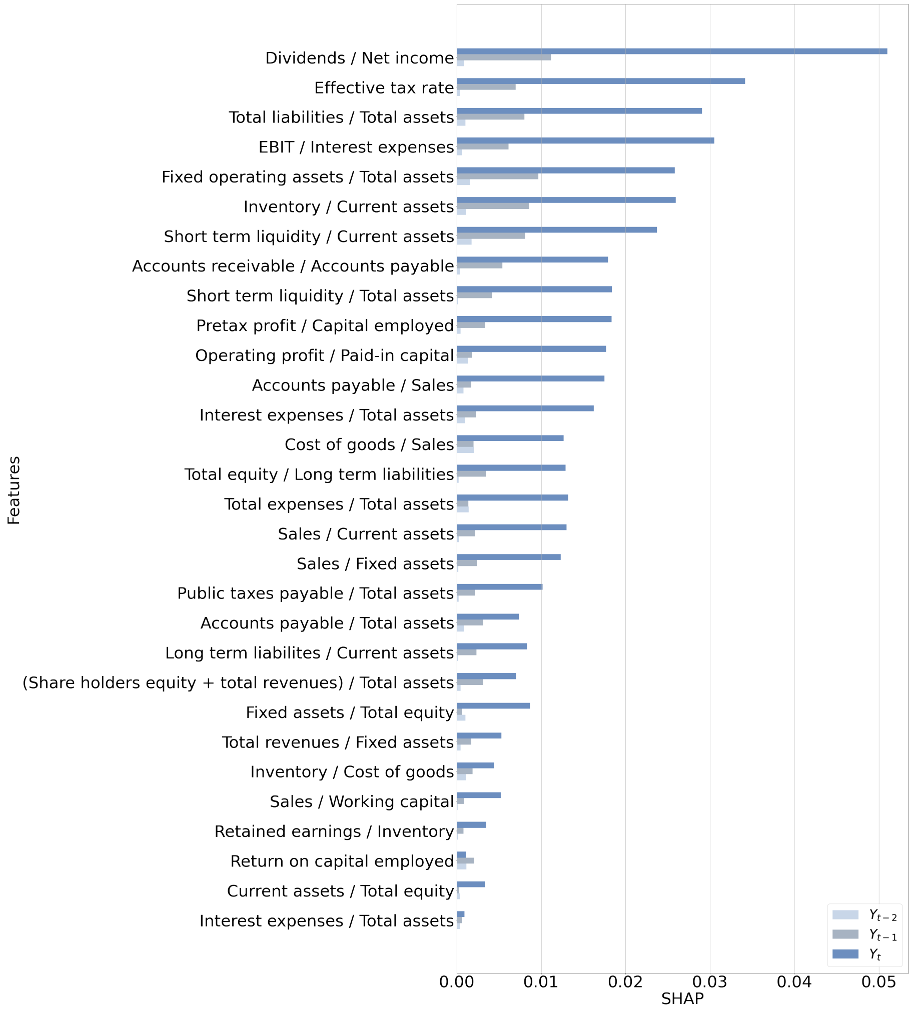


Figure B.2: The SHAP bar plot for LSTM_{3_30}

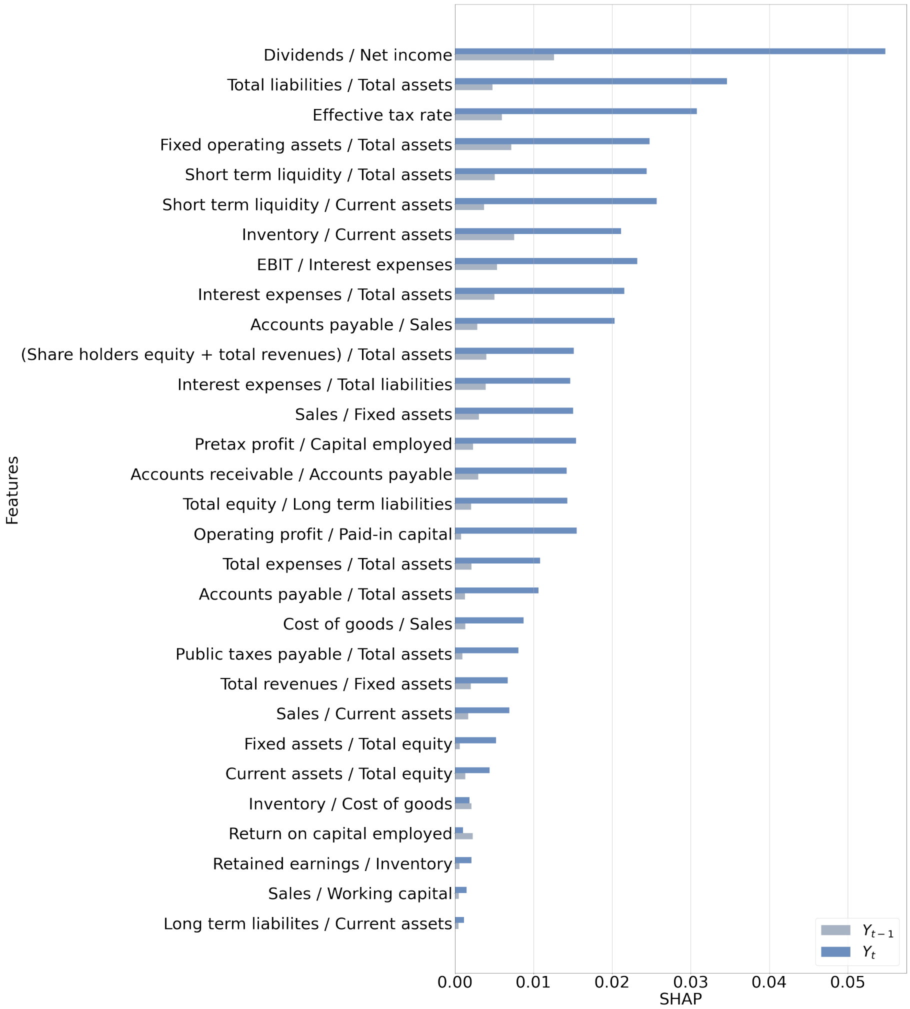
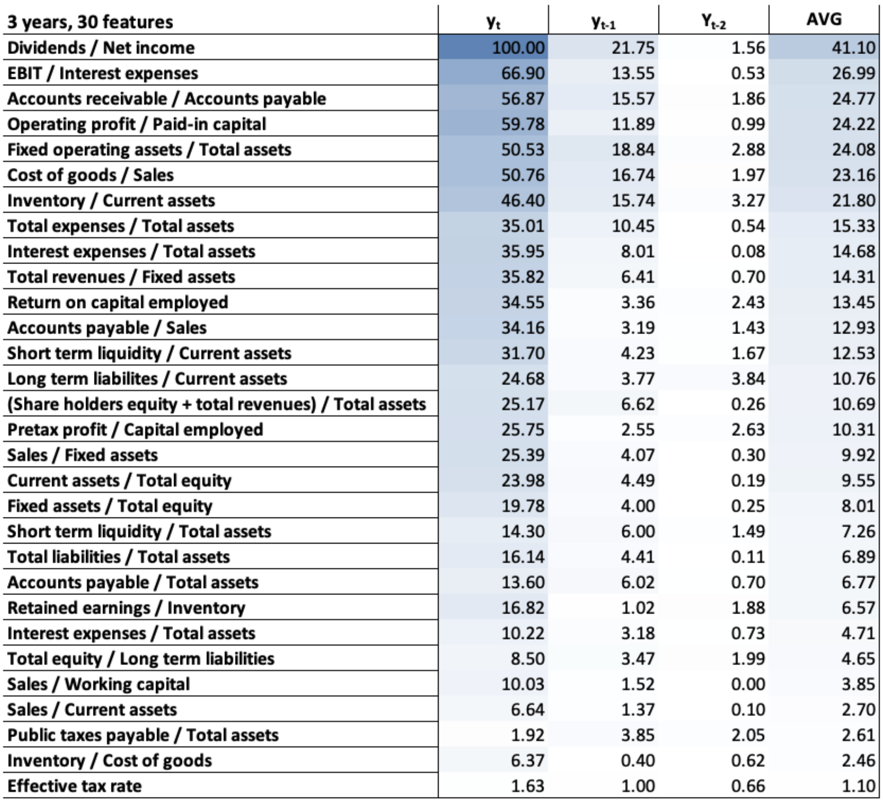


Figure B.3: The SHAP bar plot for LSTM$_{2\_30}$

Appendix C

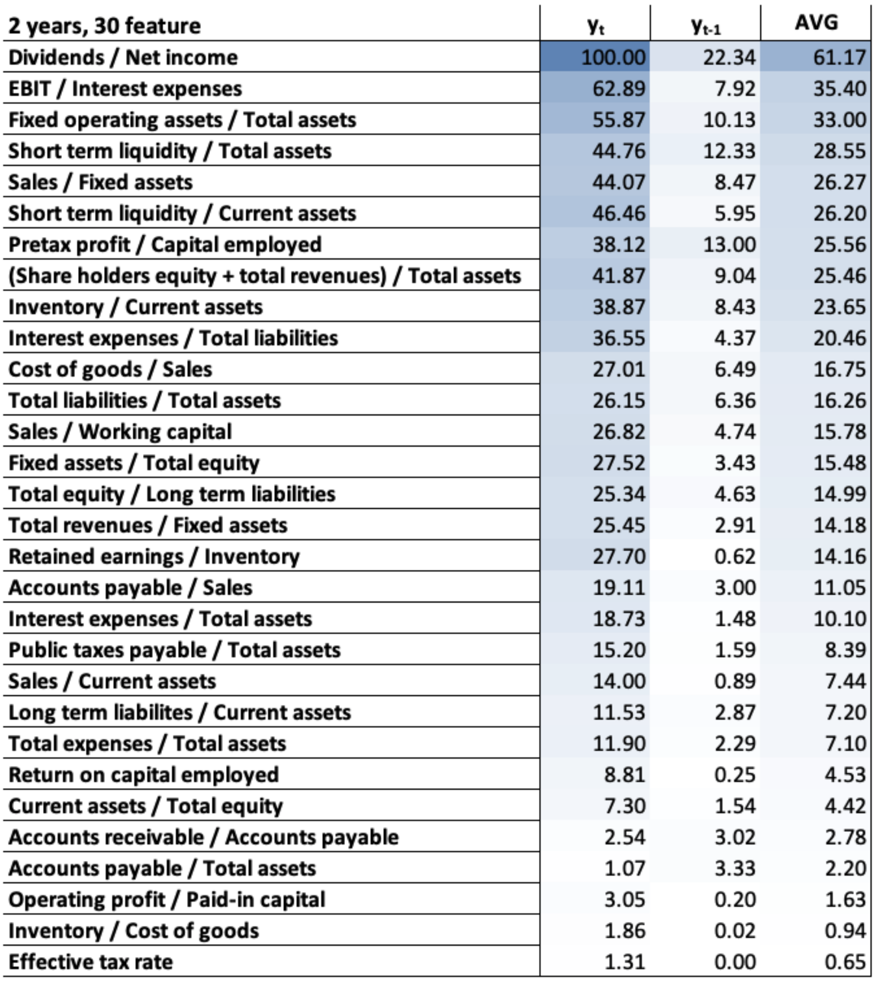
SHAP tables

The SHAP tables for the LSTM$_{3\_30}$ network and the LSTM$_{2\_30}$.



3 years, 30 features	y_t	y_{t-1}	Y_{t-2}	AVG
Dividends / Net income	100.00	21.75	1.56	41.10
EBIT / Interest expenses	66.90	13.55	0.53	26.99
Accounts receivable / Accounts payable	56.87	15.57	1.86	24.77
Operating profit / Paid-in capital	59.78	11.89	0.99	24.22
Fixed operating assets / Total assets	50.53	18.84	2.88	24.08
Cost of goods / Sales	50.76	16.74	1.97	23.16
Inventory / Current assets	46.40	15.74	3.27	21.80
Total expenses / Total assets	35.01	10.45	0.54	15.33
Interest expenses / Total assets	35.95	8.01	0.08	14.68
Total revenues / Fixed assets	35.82	6.41	0.70	14.31
Return on capital employed	34.55	3.36	2.43	13.45
Accounts payable / Sales	34.16	3.19	1.43	12.93
Short term liquidity / Current assets	31.70	4.23	1.67	12.53
Long term liabilites / Current assets	24.68	3.77	3.84	10.76
(Share holders equity + total revenues) / Total assets	25.17	6.62	0.26	10.69
Pretax profit / Capital employed	25.75	2.55	2.63	10.31
Sales / Fixed assets	25.39	4.07	0.30	9.92
Current assets / Total equity	23.98	4.49	0.19	9.55
Fixed assets / Total equity	19.78	4.00	0.25	8.01
Short term liquidity / Total assets	14.30	6.00	1.49	7.26
Total liabilities / Total assets	16.14	4.41	0.11	6.89
Accounts payable / Total assets	13.60	6.02	0.70	6.77
Retained earnings / Inventory	16.82	1.02	1.88	6.57
Interest expenses / Total assets	10.22	3.18	0.73	4.71
Total equity / Long term liabilities	8.50	3.47	1.99	4.65
Sales / Working capital	10.03	1.52	0.00	3.85
Sales / Current assets	6.64	1.37	0.10	2.70
Public taxes payable / Total assets	1.92	3.85	2.05	2.61
Inventory / Cost of goods	6.37	0.40	0.62	2.46
Effective tax rate	1.63	1.00	0.66	1.10

Figure C.1: SHAP table for LSTM$_{3\_30}$



2 years, 30 feature	y_t	y_{t-1}	AVG
Dividends / Net income	100.00	22.34	61.17
EBIT / Interest expenses	62.89	7.92	35.40
Fixed operating assets / Total assets	55.87	10.13	33.00
Short term liquidity / Total assets	44.76	12.33	28.55
Sales / Fixed assets	44.07	8.47	26.27
Short term liquidity / Current assets	46.46	5.95	26.20
Pretax profit / Capital employed	38.12	13.00	25.56
(Share holders equity + total revenues) / Total assets	41.87	9.04	25.46
Inventory / Current assets	38.87	8.43	23.65
Interest expenses / Total liabilities	36.55	4.37	20.46
Cost of goods / Sales	27.01	6.49	16.75
Total liabilities / Total assets	26.15	6.36	16.26
Sales / Working capital	26.82	4.74	15.78
Fixed assets / Total equity	27.52	3.43	15.48
Total equity / Long term liabilities	25.34	4.63	14.99
Total revenues / Fixed assets	25.45	2.91	14.18
Retained earnings / Inventory	27.70	0.62	14.16
Accounts payable / Sales	19.11	3.00	11.05
Interest expenses / Total assets	18.73	1.48	10.10
Public taxes payable / Total assets	15.20	1.59	8.39
Sales / Current assets	14.00	0.89	7.44
Long term liabilites / Current assets	11.53	2.87	7.20
Total expenses / Total assets	11.90	2.29	7.10
Return on capital employed	8.81	0.25	4.53
Current assets / Total equity	7.30	1.54	4.42
Accounts receivable / Accounts payable	2.54	3.02	2.78
Accounts payable / Total assets	1.07	3.33	2.20
Operating profit / Paid-in capital	3.05	0.20	1.63
Inventory / Cost of goods	1.86	0.02	0.94
Effective tax rate	1.31	0.00	0.65

Figure C.2: SHAP table for LSTM_{2_30}

Appendix D

List of all features

The list of all features used in the neural networks for this thesis.

Table D.1: List of all features

Number	Feature
1	(inventory + accounts receivables) / total equity
2	(long-term liability + total equity) / fixed assets
3	account receivable / sales
4	quick assets / current liabilities
5	(quick assets / current liabilities) * (operating profits / interest expenses)
6	net income / total equity
7	EBITDA / total liabilities
8	total equity / total liabilities
9	short-term liquidity as a percentage of the capital employed
10	short-term liquidity / sales
11	short-term liquidity / current liabilities
12	short-term liquidity / total assets
13	sales / current assets
14	current assets / total equity
15	current assets / sales
16	current assets / total assets (net liquid assets / total assets)
17	current liabilities / current assets
18	current liabilities / total equity
19	current liabilities / total liabilities
20	current liabilities / sales
	Continued on next page

Number	Feature
21	total liabilities / total assets
22	accounts receivable / accounts payable
23	operating profit / (operating profit - interest expense)
24	EBIT / total assets
25	EBITDA / interest expense
26	effective tax rate
27	total equity / total assets
28	total equity / long-term liabilities
29	sales / total equity
30	pre-tax profit / capital employed
31	financial expenses / sales
32	EBIT / sales
33	sales / fixed assets
34	fixed assets / total assets
35	fixed assets / total equity
36	intangibles / total assets
37	interest expenses / total revenues
38	interest-bearing debt / total equity
39	inventory / current liability
40	inventory / working capital
41	investment turnover (sales / (total equity + total liabilities))
42	total liabilities / total equity
43	long-term liability / current assets
44	net income / stockholders equity (return on shareholder's equity)
45	net income / sales
46	(total revenues - sales) / total revenues
47	total equity / fixed assets
48	total equity / sales
49	no-credit interval
50	dummy; one if total liability exceeds total assets
51	operating expenses / sales
52	short-term liquidity / total liabilities
53	operating profit / total revenues
54	operating profit / paid-in capital
55	operation asset / total asset
	Continued on next page

Number	Feature
56	personnel costs / added value
57	pre-tax net profit / paid-in capital (ordinary income / stockholder's equity)
58	net income / total revenues
59	profits / net working capital
60	quick assets / sales
61	quick assets /total assets
62	earnings after tax and interest charge / net capital employed
63	current liabilities / earnings before tax and interest charge
64	retained earnings / sales
65	retained earnings / total assets
66	return on debt (earnings / total liabilities)
67	net income / total assets
68	total revenues / fixed assets
69	total revenues / total assets
70	total revenues / net working capital
71	sales / total assets
72	total assets / total revenues
73	total expenses / total assets
74	total revenues / total expenses
75	working capital / current liabilities
76	working capital / sales
77	working capital / total assets
78	working capital / total equity
79	dummy; one if paid-in equity is less than total equity
80	working capital / total revenues
81	accounts payable / total assets
82	public taxes payable / total assets
83	EBIT / total liabilities
84	(non-interest expenses - salary) / total assets
85	(share holders equity + total revenues) / total assets
86	sales / working capital
87	short-term liquidity / current assets
88	cost of goods sold / inventory

Continued on next page

Number	Feature
89	cost of goods / sales
90	(current assets - short-term liquidity) / total assets
91	current assets / common shareholder’s equity
92	current liabilities / total assets
93	dividends / net income
94	working capital / long-term liabilities
95	working capital / operational expenditure
96	EBIT / total tangible assets
97	financial expenses / sales
98	fixed assets / (stockholder’s equity + long-term liabilities)
99	(sales - cost of goods sold) / sales
100	income gearing
101	intangible assets / sales
102	interest expenses / total liabilities
103	interest expenses / total expenses
104	interest income / interest expenses
105	interest income / total assets
106	inventory / cost of goods
107	inventory / current assets
108	inventory / sales
109	long-term liabilities / total equity
110	long-term liabilities / total assets
111	sales / tangible assets
112	net income / gross profit
113	net income / total capitalization
114	net quick assets / inventory
115	total equity / (total equity + long-term liabilities)
116	non-interest expenses / operating profit
117	total revenues / sales
118	ordinary income / total equity
119	ordinary income / ordinary expenses
120	pre-tax profit / sales
121	pre-tax profit / total assets
122	owners equity / total assets
123	payable / current liabilities

Continued on next page

Number	Feature
124	payables / inventories
125	retained earnings / inventory
126	retained earnings / tangible assets
127	return on capital employed
128	return on net fixed assets
129	salary / total assets
130	sales / short-term liquidity
131	sales / inventories
132	sales / receivables
133	sales / total tangible assets
134	interest bearing debt / total liabilities
135	share of labour costs
136	(short-term assets - total liabilities) / total assets
137	solvency ratio
138	sales / stock holders equity
139	(total revenues + interest income) / total expenses
140	interest expenses / total assets
141	operating expenses / total assets
142	tales / assets employed
143	EBITDA / total assets
144	operating profit / total assets
145	operating profit / sales
146	(current liabilities - short-term liquidity) / total assets
147	accounts payable / sales
148	retained earnings / current liabilities
149	(total equity - intangible assets) / (total assets - intangible assets - short-term liquidity)
150	EBIT / interest expenses
151	accounts receivables / total liabilities
152	profit before tax/current liabilities
153	current assets/total liabilities
154	log(age in years)
155	log(total assets)
156	log(financial expenses)

Aleksander Kaspersen & Olav Lindemark

Interpretable Deep Learning for Bankruptcy Prediction

NTNU
Norwegian University of
Science and Technology