Prediction of Bank Failure using Deep Learning

Author

UMER JAVAID

Regn Number

00000173067

Supervisor

DR. HASAN SAJID

DEPARTMENT

SCHOOL OF MECHANICAL & MANUFACTURING ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

MAY, 2019

Prediction of Bank Failure using Deep Learning

Author

UMER JAVAID

Regn Number

00000173067

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Robotics and Intelligent Machine Engineering

Thesis Supervisor:

DR. HASAN SAJID

Thesis Supervisor's Signature: ____

DEPARTMENT

SCHOOL OF MECHANICAL & MANUFACTURING ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY,

ISLAMABAD

MAY, 2019

FORM TH-4

National University of Sciences & Technology

MASTER THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by: (Student Name & Regn No.) Umer Javaid (00000173067)
Titled: Prediction of Bank Failure using Deep Learning be accepted in partial fulfillment of the requirements for the award of Masters in Robotics and Intelligent Machines Engineering degree. (Grade ________)

Examination Committee Members

1. Name: Dr. Yasar Ayaz Signature: ________________

2. Name: Dr. Muhammad Naveed Signature: ________________

3. Name: Dr. Jawad Khan Signature: ________________

Supervisor's name: Dr. Hasan Sajid Signature: ________________

Date: ____________________

____
Head of Department

____
Date

COUNTERSINGED

Date: ___________

____
Dean/Principal

Declaration

I certify that this research work titled "*Prediction of Bank Failures using Deep Learning*" is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources it has been properly acknowledged / referred.

Signature of Student

UMER JAVAID

00000173067

Plagiarism Certificate (Turnitin Report)

This thesis has been checked for Plagiarism. Turnitin report endorsed by Supervisor is attached.

Signature of Student
UMER JAVAID
Registration Number
00000173067

Signature of Supervisor

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS thesis written by Mr./Mrs. Umer Javaid (Registration No. 00000173067), of Department of Robotics and Intelligent Machines Engineering (SMME) (School/College/Institute) has been vetted by undersigned, found complete in all respects as per NUST Statutes / Regulations, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in this dissertation.

Signature: ______________________________

Name of Supervisor: Dr. Hasan Sajid

Date: ______________________________

Signature (HOD): ______________________________

Date: ______________________________

Signature (Dean/Principal): ______________________________

Date: ______________________________

Copyright Statement

Acknowledgements

I am thankful to my Creator Allah Subhana-Watala to have guided me throughout this work at every step and for every new thought which you set up in my mind to improve it. Indeed, I could have done nothing without your priceless help and guidance. Whosoever helped me throughout the course of my thesis, whether my parents or any other individual was your will, so indeed none be worthy of praise but you.

I am profusely thankful to my beloved parents who raised me when I was not capable of walking and continued to support me throughout in every department of my life.

I would also like to express special thanks to my supervisor Dr. Hasan Sajid for his tremendous support and cooperation. Each time I got stuck in something, he came up with the solution. Without his help I wouldn't have been able to complete my thesis. I appreciate his patience and guidance throughout the whole thesis.

I would also like to thank Dr. Yasar Ayaz, Dr. Muhammad Naveed and Dr. Muhammad Jawad Khan for being on my thesis guidance and evaluation committee and express my special thanks to Dr. Yasar Ayaz for his help.

I am also thankful to Sadia Ghazal for replenishing my confidence, whenever I needed it, Mehreen Ali for assisting during the writeup process, Ahmad Mansoor and Tayyaba Naz for their continual moral support. I would like to mention Aymeric Damien for making his source code (https://github.com/aymericdamien) available, it's a great way to get started on TensorFlow.

Finally, I would like to express my gratitude to all the individuals who have rendered valuable assistance to my study.

Dedicated to my family; my caring father Prof. Javaid Ali (late), my loving mother and my supporting brother without whom I wouldn't have been able to complete this.

I wish you were here father, to see this. It's funny, how I thought completing it would make me happy and now when it's finally done, it's not the same without you around. I wish I could have done this a year earlier, so you would have been there to see it and how proud it would have made you feel.

In the end it all comes down to regrets… May your soul rest in peace

Abstract

Banking sector at the core is responsible for holding of financial assets in any economy. Bank Failure has far greater effect on the overall economy of a country than any other businesses. It can quickly spill over to other banks and financial institutions and therefore has a multiplying effect. To avoid such scenarios, rigorous regulations have been put in place along with technology to monitor, track and forecast critical parameters. Various statistical techniques and machine learning approaches have been widely adopted in this context. Banks hire domain experts, who along with their expertise exploit these tools to make decisions and recommend actions to prevent bank failure. Despite such tools and expertise, bank failure has occurred from time to time due to complexity of the problem since it is hard to generalize all the knowledge. Lately, with success of AI across different domains, financial institutions and banks have started to adopt much powerful AI methods to replace old methods. In continuation of the modernization effort, this paper proposes a novel deep recurrent neural network for bank failure prediction. More specifically we propose a four-layer recurrent network with Long Short-Term Memory cells. To validate the proposed algorithm, we collected data of 1139 banks from G7 countries and Australia around global financial crises from 2003 to 2013. In total we have collected 59 ratios and variables over eleven years for each of the bank. The proposed algorithm is compared against baseline implementations of widely adapted SVM and Logistic Regression methods. Empirical results demonstrate the superiority of the proposed approach. The paper concludes with a detailed study of effect and role of different parameters towards bank failure.

Key Words: *Bank failure, Long Short-Term Memory, Recurrent Neural Networks, Deep Neural Network*

Table of Contents

List of Figures

List of Tables

CHAPTER 1: INTRODUCTION

The research work in this dissertation has been presented in two parts. The research work in this dissertation has been presented in two parts. **CHAPTER 2:** contains an overview of the work done in predicting bank failure, **CHAPTER 3:** describes the dataset and its properties, **CHAPTER 4:** describes the proposed LSTM network, comparison with existing techniques and advantages over them, **CHAPTER 5:** comprises of the experimental setup used in this study, results and comparisons with the baseline implementations and **CHAPTER 6:** finally concludes the dissertation with limitation of the study and future work.

1.1 Background, Scope and Motivation

Banks stand at the core of any economy's financial system. They are the creators of capital by becoming the main financial intermediary between depositors and borrowers. A banking crisis occurs when a bank is unable to meet the demands of its creditors due to a decline in the value of its assets below the value of its liabilities. In other words, the bank does not remain liquid enough to fulfill its liabilities; thus, it either borrows from other stable banks or is forced to sell its assets at a lower price to meet the demands of depositors. The bank fails when it has become too illiquid to operate, hence it is taken over by the state or federal banking regulatory body and is closed down. This has a multiplying effect, as the announcement of bank failure creates a panic among depositors, who, fearing more loss, take out their deposits from other banks too. This increase in cash withdrawals creates a risk of making other banks insolvent too. Consequently, the failure of banks has a fatal impact and is more catastrophic than any other business failing. If the crisis prevails, it has pernicious effects not only on the local economy but also ripples to international economies. As such, every economy subjects its banks to rigorous regulation policies and contracts to ensure that they remain solvent enough to run smoothly and to preserve confidence in the financial system of the economy.

Given the importance of the problem numerous solutions have been developed to predict the failure before it occurs as an "early warning system". Traditionally such analyses are performed by domain experts. Hiring a domain expert, generally twice a year, can be very costly. Even if a bank is willing to bear this cost the chance of human error still exists, after all the experts also use previous data to analyze trends and predict future outcomes. During the last few decades, a lot of research is being done in this domain, many statistical and machine learning solutions have been proposed, each having its own pros and cons.

Existing methods in general suffer due to multiple shortcomings that lead to poor prediction accuracy. The choice and selection of input features is subjective and can vary from one domain expert to another. This results in missing out important information in the form of features and parameters. Existing methods are relatively simple to model the complex non-linear variations of parameters overtime and correlation among them. They do not capture the temporal variation of parameters instead they map the input features to output labels. The nature of the problem demands for methods that can not only account for temporal variation but also the variation in length of sequences.

To address the aforementioned challenges, we propose a deep Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) cells. Similar to Artificial Neural Networks (ANN), RNNs map input features to the target labels. However, recurrent neurons (basic unit of a RNN) are a function of both current state (input at time step t) and previous state (output of another neuron at time step t-1), whereas artificial neurons take current state (input at time step t) as an input only. Connecting multiple recurrent neurons in a chain results in a form a memory since the output of a neuron at times step t is a function of all inputs from previous time steps. This property of recurrent networks makes them ideal for time-series data and ability to handle sequences of arbitrary length. Architectures with multiple layers of RNNs are possible to learn more complex representations. The inherent characteristics of LSTM network (RNNs with LSTM unit cell) is the key in addressing problems specific to bank failure prediction: objective selection of input features and spatio-temporal modeling of input features. First, the network takes all the features or data as input and automatically learns to assign more weightage to important features to correctly predict the target label. The end-to-end approach automatically caters for any bias in selection of features. Secondly, it can create additional features from input data/features to improve the discrimination power of the classifier. Lastly, it can not only capture correlation among features but also the temporal variations of the features with capability to handle variable number of timestamps, a key shortcoming of previous methods. The proposed approach is validated on a new dataset, which we contribute for research community. It comprises of a variety of financial features and ratios based on CAMELS rating system. The data is collected over a period of 11 years (2003-2013) during the recent financial crisis. To the best of our knowledge no such dataset exists that provide variation of features overtime. To compare our approach on this dataset, we provide baseline implementations of commonly used machine learning methods in this domain: SVM and Logistic Regression. The proposed approach outperforms these methods. Lastly, we provide a detailed study of the effect of the different features that contribute towards bank failure.

CHAPTER 2: REVIEW OF POPULAR TRADITIONAL APPROACHES

There have been many researchers who analyzed, used and compared different mechanisms that help predict bank failures. The first research in the area of bankruptcy was conducted by Altman (1968) [1]. Following that, during the last five decades, the research methodology has changed a lot and has varied from statistical models to intelligence systems, but the variables and the idea of using financial ratios to predict bankruptcy is, more or less, the same.

1.2 Statistical Techniques

Over many years, discriminant Analysis (DA) has been one of the most used statistical technique in determining bank failures (Karels & Prakash (1987) [2]; Haslem et al. (1992) [3]). DA is divided into three subcategories i.e. Linear, multivariate and Quadratic. DA is used to analyze cross-sectional data. Time series data is often analyzed using hazard or duration analysis models. In order to perform DA, normal distribution of regressors is required. Maximum likelihood method of Logit is applied when regressors are not normally distributed. Logit is an abbreviation of Logistic Regression, which is a predictive analysis technique which uses a model function (known as the logistic function) to model a variable which is binary dependent.

According to West (1985) [4] Factor analysis combined with Logit estimation is valuable in assessing banks' working conditions. West findings also suggested that the factors identified as significant variables, determining banks' operating conditions, closely resembles CAMELS ratings. CAMELS rating system was developed for the classification of bank's overall condition. The categories addressed by this rating system are (C)apital adequacy, (A)ssets, (M)anagement capability, (E)arnings, (L)iquidity and (S)ensitivity.

EWS have been widely used by central banks to monitor bank risks, on the other hand, "Federal Deposit Insurance Corporation Improvement Act of 1991" made it mandatory for banks to lead nearby examinations of bank hazard each 1-1.5 year. The maintainers use CAMELS rating framework to show each bank's wellbeing and soundness. Davis & Karim (2008a) [5] applied statistical intelligence techniques in assessing bank crises. The study compared logistic regression (Logit) and Signal Extraction in Early Warning Systems (EWS). The findings suggest that the choice of estimation models has a grave impact on the performance of variables and hence the crises predictability. Logit models have a better predictability in global EWS while Signal Extraction is a good predictor in country specific EWS. In another study, Davis & Karim (2008b) [6] tested Logit and binomial tree based EWS which helped predict bank failures in US and UK. The results suggested that the Logit performs better than rest of the techniques.

Building upon EWS, an Integrated Early Warning System (IEWS) was proposed by Canbas et al. (2005) [7]. IEWS unites the DA, Logit, Probit and Principal Component Analysis (PCA) models to foresee bank failures. The system initially uses PCA to detect the financial components that best represents the condition of banks subsequently DA, logit, Probit regression models are applied. IEWS shows a better predictability power than most single models used in literature works.

1.3 Intelligent Approaches

Other studies have applied intelligence modelling techniques in operational research to predict bank failures and crises. Amongst the most widely used intelligence technique is Neural Networks (NN). NN models contain mathematical and algorithmic substances that portray biological neural networks of the human nervous system. Some examples include Celik & Karatepe (2007) [8], who utilized artificial neural network models to forecast crises, and Alam et al. (2000) [9], whose study used fuzzy clustering and self-organizing neural network in identifying failed banks.

A research conducted by Boyacioglu et al. (2009) [10] compares numerous NN, Support Vector Machine (SVM), Multivariate Discriminant Analysis, Cluster Analysis and Logit regression analysis applied in CAMELS setting to detect bank failures in Turkey. The results indicated that Multivariate Discriminant Analysis and Logit regression analysis are better failure predicting models among all others.

A multilayer NN model, known as Back-Propagation Neural Network (BPNN) model. was used by Tam (1991) [11] to successfully predict Texas bank failures almost one to two years before the collapses. BPNN is the most commonly used classification and prediction method as it outperforms other models. The first and last layers comprise of input and output units, while the middle layer consist of hidden units . The unique key feature of the BPNN model is that the errors generated by the hidden layers are calculated by “back propagating” the errors of the output sent by the corresponding layer. Tam (1991) [11] used CAMELS variables in his research and concluded that BPNN outperformed K-nearest neighbor, DA and Logit technique, in predicting bank failures accurately.

Tam & Kiang (1992) [12] applied “linear discriminant analysis (LDA), Logit, K-Nearest Neighbour, Interactive Dichotomizer 3 (ID3), feedforward NN and BPNN” in predicting bank failure. Amongst all the models applied, BPNN outperformed all models for one-year prior samples while LDA outperformed the rest for two years prior samples. However, BPNN outperforms all, in both one- and two-year prior samples for holdout samples and in jackknife method. They concluded their study by indicating that NN outperforms DA method.

A study was conducted by Bell (1997) [13] for predicted bank failures using Logit and BPNN models. His findings show that neither Logit nor BPNN model is superior to one another when it comes to predictability. The methodology applied twelve input nodes, six hidden nodes and one output node in BPNN. Concluding that BPNN is better where complex decisions are to be made.

Swicegood & Clark (2001) [14] on the other hand found that the identification of underperforming banks can be done better with BPNN. The study compared DA, PNN and human judgment in bank failure prediction.

1.4 Logit and Trait Recognition Approaches

Another approach to predict collapses are Trait Recognition models. They are developed from different distribution segments for each variable and based on the interactions with one or more variables in the distribution segments. Two sets of discriminators are used; safe and unsafe traits; to anticipate bank failures by classifying each bank under one of the two discriminators. Trait recognition is able to analyze complex correlation of variables. The strongest point of this approach is that it uses cut off point for each variable, indicating a threshold, where all failed banks are situated underneath it and the banks that survived above it.

Kolari et al. (2002) [15] considered a large set of US banks and applied EWS based Logistic Regression and Trait Recognition methods. The Logit model was able to accurately classify 96% of the banks, one-year and two-year prior to their shut down. Trait Recognition model, on the other hand shows an accuracy of 100%. According to their analysis, in terms of Type 1 and Type 2 errors, Trait Recognition model performs better than Logit model.

Logit model and Trait Recognition approach were also used by Lanine & Vander Vennet (2006) [16] to predict bank failures in Russia. The two models were tested based on their predictive accuracy. Trait Recognition approach outperformed Logit in holdout and original samples. The results imply that liquidity, asset quality and capital adequacy are important in determining bank failures.

There are some techniques originating from machine learning such as Decision Trees (DT), which apply "recursive partitioning algorithm” to organize rules on a given data set. Algorithms like classification and regression trees (CART) can be used to solve forecasting problems. Furthermore, set of rules derive the development of a binary decision tree in order to be used for accurate classification of banks. For this, algorithms such as “CHAID (chi-square automatic interaction detection)”, “CART”, “C4.5” and “C5.0” have been used (Marais et al. (1984) [17] and Frydman et al. (1985) [18]).

1.5 Other Approaches

Rough Set techniques on the other hand, models partial data based on a concept specified by Pawlak (1982) [19]. It is a mathematical method which apples estimation of rough objective into predefined classes to be examined. (see Greco et al. (1998) [20] for more details. Ahn et al. (2000) [21] integrated rough set theory and artificial neural networks to forecast failures. This proposed hybrid model outperformed discriminant analysis models and neural network models.

A much simpler technique for accuracy in predictions is Case base Reasoning (CBR) which allows failure predictions based on the past experiences. This technique is like the psychological procedure people pursue to take care of issues instinctively. CBR comprises of four steps. First, retrieve similar cases. Second, reuse the cases to solve the problems. Third, revise proposed solution, if possible. Fourth, hold new arrangement as part of new case.

Nearest Neighbour technique is composed of classification of an object (bank) based on the object (bank) in the class of its nearest neighbour. The objects can be tested for being random, clustered or regularly distributed. Banks are assigned to the most common amongst its K nearest neighbours’ classes i.e. survived or failed.

An additional comparative study conducted by Zhao et al. (2009) [22] suggested that the several factors used to predict bank failure using Logit, DT, NN and K-NN models, the importance of model choice in terms of its explanatory power of the predictors stands out.

1.5.1 Support Vector Machine

Support Vector Machine (SVM) technique is derived from the Structural Risk Minimization (SRM) principle. SRM principle originates from computational learning theory coined by Vapnik (2013) [23]. Input data in SVM is organized from two sets of vectors in multi-dimensional space. SVM applies a specialized linear model and ideal separating hyperplane to achieve extreme division between the two classes. Many have applied this approach amongst them are Vapnik (2013) [23], Boyacioglu et al. (2009) [10], Chen & Shih (2006) [24] and Huang et al. (2004) [25]. Shin et al. (2005) [26] suggest SVM technique outperforms BPNN in predicting bankruptcy of corporations and has a higher accuracy level.

1.5.2 Data Envelopment Analysis

Data Envelopment Analysis (DEA) measures the efficiencies of organizations or decision-making units (DMU) which applies linear programming to observe devoured and yields delivered by DMU i.e. bank branches. DMU builds an effective creation wilderness dependent on best practices. Each DMU productivity is estimated against computed frontier. The effectiveness is determined by the weighted sum of all outputs and weighted sum of all inputs. The weights achieve Pareto optimality for each DMU.

DEA model was also used to measure marketability and profitability efficiency in large banks by Luo (2003) [27]. The study finds “marketability inefficiency” makes tremendous issues for banks compared to “profitability inefficiency”. In predicting bank failures, the likelihood is reduced when banks are profitable.

DEA and network DEA (NDEA) technique was used to analyze profit efficiency in United Arab Emirates banks by Avkiran (2009) [28]. Standard DEA model does not identify specific sources of inefficiency. Proficiency measures from stochastic DEA models don't consider “statistical noise” also, measurement error is ignored.

Kao & Liu (2004) [29] formulated DEA model to evaluate bank performance. Their study makes advance predictions based on banks in Taiwan. The model was able to predict two bank failures in advance, adding credibility to it.

Another framework Bayesian approach, a statistical framework was utilized with stochastic DEA make induction on proficiency scores. The results from Tsionas & Papadakis (2010) [30] suggest lion's share of Greek banks work near market best practices.

1.6 Hybrid Approaches

Multicriteria decision technique; UTilites Additives DIScrim-inants (UTADIS), was developed by Kosmidou & Zopounidis (2008) [31] to predict bank failure. UTADIS works well with “ordinal classification problem” and is not delicate to statistical problems predominantly on the grounds that the additive utility function is not performed through statistical methods instead its done using mathematical linear programming technique. The conclusion derived is UTADIS predicts bank failures four years prior to the occurrence. UTADIS also performs better than traditional multivariate data analysis techniques.

Multicriteria Decision Aid (MCDA) method, used to determine credit ratings and bank soundness, performs superior to conventional multiple discriminant analysis. Gaganis et al. (2006) [32] applied MCDA model using the UTADIS method to categorize banks depending on their soundness. The model was developed through cross-validation procedure. The results highlight that the most significant criteria in classifying bank soundness is based on capitalization, asset quality and banks’ operating market. Bank performance is also defined by profitability and efficiency. UTADIS has the best classification accuracy in comparison to DA and logit. Pasiouras et al. (2007) [33] tested MCDA model to check if it can emulate credit rating model of Fitch. Their findings suggest that the most significant financial ratios were equity to customer and short-term funding, net interest margin and return on average equity. In terms of non-financial factors number of shareholders, number of subsidiaries and banking environment were significant. In comparing MCDA with ordered Logit and DA model, MCDA has the closest resemblance to Fitch credit ratings.

Financial crises have also been predicted with Analytic Network Process (ANP) framework. It is a multiple criteria decision-making model. Niemira & Saaty (2004) [34] findings showcased that ANP framework in comparison to traditional models is more flexible and comprehensive and thus a good model to forecast crises. ANP framework also reduces judgmental forecast errors because of its structure construction.

There are hybrid methodologies also researched and used over time. According to Ravi & Pramodh (2008) [35], hybrid models that combine Principal Component Neural Network (PCNN) and several other failure prediction models, outperform other classifiers applied across literature works. In PCNN framework, the hidden layer is replaced by 'principal component layer' and a few selected components carry out the hidden nodes' role.

Soft Computing technique also falls under the hybrid methodologies, which uses both intelligence systems and statistical techniques. Computational techniques are applied to model and analyze complex phenomenon. Where hard computing deals with exact algorithms and calculations, soft computing is based on estimated computations, subjective decision making and trial and error reasoning. This method replicates the cognitive process of human minds. See Back et al. (1996) [36], Jo & Han (1996) [37] and Tung et al. (2004) [38].

Fuzzy Cerebellar Model Articulation Controller model (FC-MAC) was created by Ng et al. (2008) [39]. FCMAC, based on FCMAC-CRI(S), a compositional rule of interference. The model integrates fuzzy systems and NN to create neural fuzzy networks. The network takes in financial information as input data and analyses the patterns of bank failures. This is done through fuzzy IF-THEN rules. In comparing FCMAX-CRI(S), Cox proportional hazard model and GenSoFNN-CRI(S) network model, FCMAX-CRI(S) performed better than the rest.

Minimized sum of Deviations (MSD) is a combined model of DA and linear programming (LP). Cielen et al. (2004) [40] compared the performance of DEA model, MSD and rule of induction (C5.0) on National bank of Belgium. All models correctly classified the rates of failures. However, DEA turned out to be the best model overall in terms of accuracy prediction.

CHAPTER 3: DATASET

The data was collected data from 1139 banks located in 8 major countries across the globe, which include G7 countries (Canada, France, Germany, Italy, Japan, United Kingdom and United States) and Australia. These eight countries have the largest and most advanced economies in the world and represent more than 60% of the global net wealth. Data was collected from 2003-2013, which is the time period of most recent global financial crisis. Many banks failed during this time period, which makes it ideal for bank failure study. The dataset contains a total of 12529 samples with each sample formed by 59 different parameters and ratios. The parameters/ratios were categorized using the CAMELS rating system, originally known as the Uniform Financial Institutions Rating System (UFIRS) (Council, 1996). **Table 3-1**, **Table 3-2**, **Table 3-3**, **Table 3-4**, **Table 3-5** and **Table 3-6** present these parameters and ratios for each of the categories: capital, asset, management, earnings, liquidity and sensitivity respectively.

3.1 CAMELS rating system

We will briefly discuss each category of the CAMELS rating system. Capital adequacy category is the measure of bank's capital condition. It also reflects the ability of a bank to handle the losses and prevent its operations from ceasing. Asset quality category is the measure of the efficiency of bank's investment policies and practices. It is measured by rating the risk factors the bank may face as compared to its capital earnings. Management quality category is the measure of effectiveness of top-level management personnel responsible for the successful operations of a bank and their ability to adapt and respond to the market trends. Earnings are described as the return on assets ratio. It includes the income of a bank from all the sources i.e. operations and other non-traditional sources. Liquidity is the ability of a bank to convert its assets to cash. It is determined by the ratio of cash maintained by banks to total assets. Sensitivity is described as the effects of adverse market changes on a bank such as abrupt changes are foreign exchange rates, commodities prices etc.

3.2 Challenges with the dataset

The dataset had two main challenges: imbalanced data and variable sequence length. The proportion of banks that survived (56%) and negative (banks that failed) examples is not equal.

Secondly, all the samples were not of 11 units length (from 2003-2013). For example, some of the banks failed earlier than 2013 and some banks started after 2003.

Table 3-1: List of Capital features

Global S. No.	Category	S. No.	Feature Description
1	Capital	1	capital adequacy ratio Tier 1
2		2	capital adequacy ratio Tier 2
3		3	common equity to total asset
4		4	debt to equity
5		5	retained earnings
6		6	net income and total equity to deposit and short-term funding
7		7	net income and total equity to total asset
8		8	regulatory Tier 1 capital to risk weighted asset
9		9	equity to asset

Table 3-2: List of Asset features

Global S. No.	Category	S. No.	Feature Description
10	Assets	1	allowance for loan loss
11		2	common equity to net loan
12		3	equity to net loan
13		4	gross non-performing loan to advances
14		5	non-performing loan to gross loan
15		6	Provision for loan loss
16		7	loan loss provision to average asset
17		8	loan loss provision to net interest income
18		9	Provision for loan loss to total loan

19		10	non-performing loan to net advances
20		11	non-performing loan to total equity
21		12	total loan to total asset

Table 3-3: List of Management features

Global S. No.	Category	S. No.	Feature Description
22	Management	1	business per employee
23		2	loan growth rate
24		3	management expense
25		4	profit per employee
26		5	total loan to total deposit

Table 3-4: List of Earning features

Global S. No.	Category	S. No.	Feature Description
27	Earnings	1	cost to income
28		2	dividend payment
29		3	earnings per share
30		4	interest income to interest expense
31		5	interest income to total income
32		6	non-interest expense to average asset
33		7	non-interest expense to gross income
34		8	non-interest expense to total expense
35		9	non-interest expense to total customer deposit
36		10	net interest margin to gross income
37		11	net interest margin
38		12	net interest revenue to average asset
39		13	non-interest income to total income

40		14	net income to average asset
41		15	net interest income to asset growth rate
42		16	non-interest income to non-interest expense
43		17	operating income to total asset
44		18	pre-tax income to average asset
45		19	pre-tax income to revenue
46		20	return on average asset
47		21	return on average equity
48		22	tax to earning before tax
49		23	interest expense to total expenses

Table 3-5: List of Liquidity features

Global S. No.	Category	S. No.	Feature Description
50	Liquidity	1	customer deposit to total asset
51		2	liquid asset to customer and short-term funding
52		3	liquid asset to deposit and non-deposit fund
53		4	liquid asset to short term liabilities
54		5	liquid asset to total asset
55		6	liquid asset to total deposit
56		7	net loan to total asset
57		8	Non-performing loan to total asset
58		9	total loan to customer deposit

Table 3-6: List of Sensitivity features

Global S. No.	Category	S. No.	Feature Description
59	Sensitivity	1	log of total asset

CHAPTER 4: PROPOSED APPROACH

First, we briefly discuss the theory of RNNs, key differences to popular ANNs, its relevance to financial data and the proposed method. Financial parameters and features are in general non-linear, highly correlated and temporal in nature i.e. correlation is not only valid for a single time instance (static) but also over multiple time steps. Variation in one parameter can cause other parameters to fluctuate and therefore affects the overall output or contribution towards bank's survivability. For example, if there is a positive change in the liquidity ratios of a bank it implies that the bank is refraining from investing the capital thus in the longer run it will have a negative impact on the earnings of the bank. Another example is that if the earnings of certain bank are low, it implies that the bank suffered loss or relatively lower profit in its investments. This loss has a negative impact on the capital as it is recovered from it at the end of the year; this process is known as capital erosion.

4.1 Artificial Neural Networks

To better understand we first describe ANN and RNN. ANNs are inspired by the human neurological system, which consists of neurons connected together in a mesh like configuration. ANNs consist of a directed graph like structure, where each node of the graph is analogous to a neuron and each connection is analogous to a synapse with a certain weight assigned to it. The network learns to adjust these weights for the given problem. **Figure 4-1** shows a typical ANN architecture. The neural networks have a layered architecture. Generally, there are three different layers in a neural network: input layer, hidden layer(s) and the output layer.

The non-linear nature of the data makes ANNs a decent fit for the problem as they are capable of learning complex inter-feature dependencies however vanilla ANNs don't take the temporal property of the data into consideration, thus the model trained using ANN's is unable to learn any periodic feature variations and dependencies.

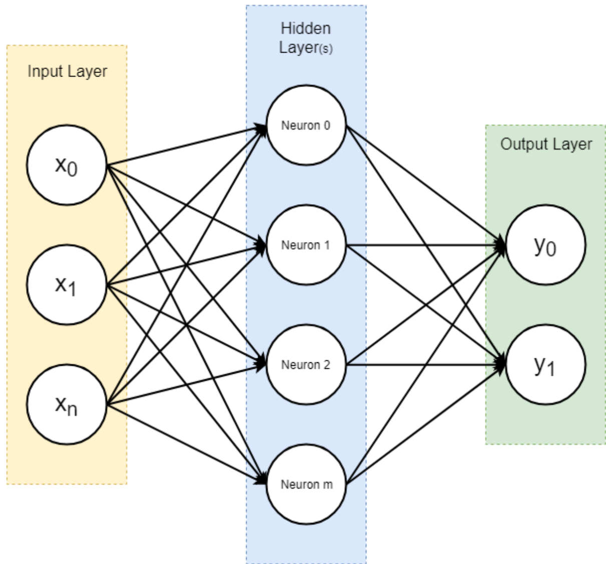

Figure 4-1: Typical ANN Architecture

4.2 Recurrent Neural Networks

A newer class of Neural Networks, namely Recurrent neural networks (RNNs), are designed to take this temporal factor into consideration. Visually RNNs are similar to ANNs, however, unlike ANNs, which are feed forwards networks, RNNs have connections pointing backwards. Figure below shows a recurrent neuron and multiple neurons connected together overtime.

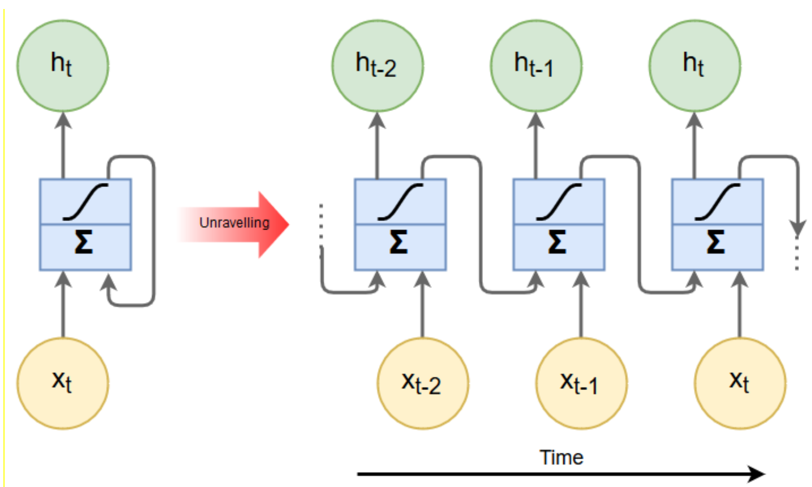

Figure 4-2: Singe RNN Cell (left), Unraveled Single RNN Cell (Right)

Figure 4-2 (left) shows a single Recurrent neuron. The neuron receives an input and produces an output, which is fed back to the recurrent neuron itself. If multiple neurons are connected together and unrolled through time as shown in **Figure 4-2** (right), it is evident that at each time step every neuron receives two inputs, comprising of an input at the current time step and an output from the previous time step. This implies that each neuron has two sets of weights one for the input and other for the output from previous time step, thereby creating a form of memory. Similar to ANNs, multiple layers can be created with single or multiple recurrent neurons **Figure 4-3**.

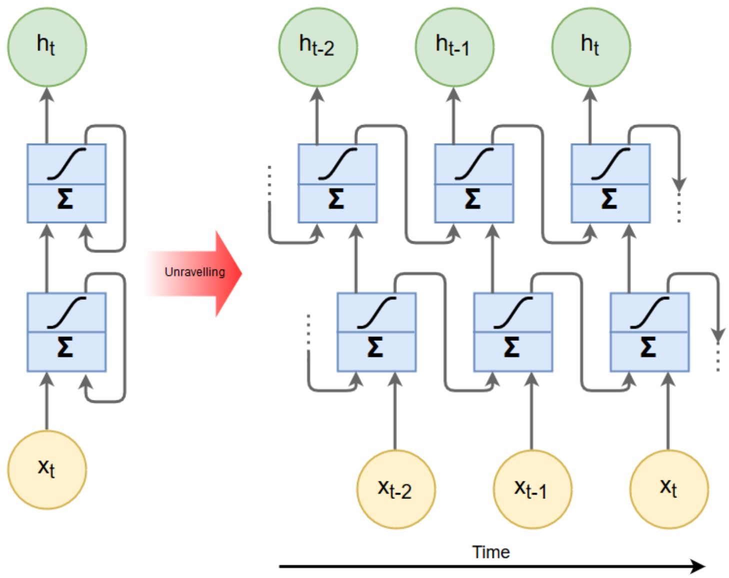

Figure 4-3: Deep Recurrent Neural Network

4.2.1 Gradient Vanishing Problem

Unfortunately, the vanilla RNNs are known to suffer from gradient vanishing problem (Hochreiter, 1998) [41] especially with networks getting deep and deep. LSTMs and GRUs are well known variants of basic RNN cell that were proposed by (Hochreiter & Schmidhuber, 1997) [42] and (Chung et al., 2015) [43] respectively to cater gradient vanishing problem by rerouting gradient through alternative path. We employ LSTM variant of RNNs in the proposed approach. Furthermore, LSTMs offer the advantage of faster convergence during training, automatic identification of important features and retention of long-term dependencies in the data. For further subtleties of LSTM, we allude pursuers to (Hochreiter & Schmidhuber, 1997) [42].

4.3 Proposed Network Architecture

We propose a 4-layer deep recurrent neural network comprising of LSTM cells **Figure 4-4**. Each cell comprises of 128 recurrent neurons. The data was normalized using equation **(4.1)** before passing it to the network. It is done to ensure that no parameter dominates the other only because of its magnitude.

$$x_{norm} = \frac{x - Min(x)}{Max(x) - Min(x)} \tag{4.1}$$

Where x_{norm} is the normalized value of feature x, $Min(x)$ is the minimum value of feature x and $Max(x)$ is the maximum value of feature x. The network takes a three-dimensional block of input with size $n \times \delta \times \beta \mid \beta, \delta \in \mathrm{N}$ where n is the number of features, δ is the number of time steps and β is the batch size hyper-parameter. The network was designed to take variable time step block of input. This was done to handle the variable sequence length challenge associated with the dataset, as mentioned in section 3. The network produces a one hot encoded vector with dimension 1×2 at each time step.

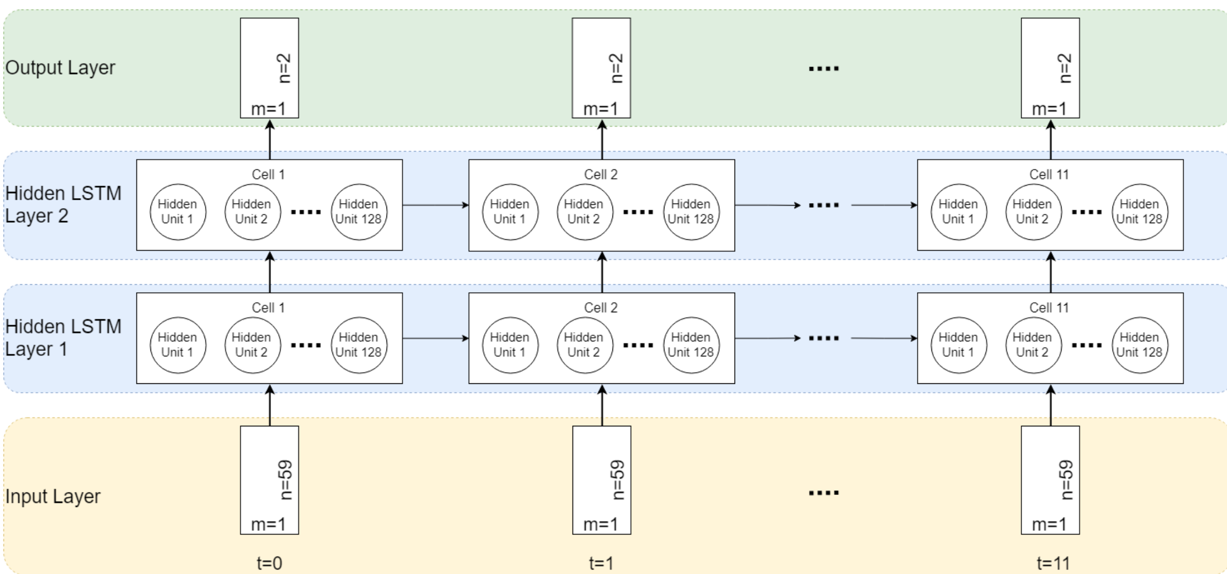

Figure 4-4: Proposed LSTM Network Architecture

4.3.1 Internal working of LSTMs

The neurons inside an LSTM network are also referred to as LSTM cells. Each cell is composed of three gates. These gates are responsible for control of flow of information. Each gate can have a value from 0 to 1, Zero means block the input value whereas 1 means let all the value in the cell.

Inside the cell the input feature vector is concatenated with the previous hidden state of the cell (equation **(4.2)**) and is passed through the gates. The 'input gate' controls the feature values that will be passed on to the cell state at the next time step (equation (4.3)). The 'forget gate' decides what to forget and the rest is passed on to the next cell state (equation **(4.4)**). At this point the internal hidden state of the cell is updated (equation **(4.6)**). The 'output gate' filters the values of the cell state i.e. outputs a necessary subset of the contents of this cell state that can generate a right prediction (equation **(4.7)**). The next hidden state is also generated at this point (equation **(4.8)**).

$$I_{t0} = [h_{t-1}, x_t] \tag{4.2}$$

$$i_{t0} = \sigma\big((W_i . I_{t0}) + b_i\big) \tag{4.3}$$

$$F_{t0} = \sigma\ (\big(W_f . I_{t0}\big) + b_f \tag{4.4}$$

$$\widetilde{C_{t0}} = tanh\big((W_C . I_{t0}) + b_C\big) \tag{4.5}$$

$$C_{t0} = (f_{t0} * C_{t-1}) + \big(i_{t0} * \widetilde{C_{t0}}\big) \tag{4.6}$$

$$O_{t0} = \sigma(W_O * I_{t0}) + b_O) \tag{4.7}$$

$$h_{t0} = O_{t0} * tanh(C_{t0}) \tag{4.8}$$

Some of the properties of a financial dataset is that is non-linear by nature, the features have a strong inter-correlation and when considering the data over a long duration temporal factor also gets involved. LSTMs contain non-linear activation functions (ReLU and Sigmoid), enabling it to perform better on the financial dataset. The complex inter-feature dependencies are handled by the proposed deep layered architecture, instead of mapping the features directly to the outputs these features are converted into complex features for the hidden layers of the network. LSTMs being part of the recurrent neural networks are designed for temporal datasets. The recurrent links on the neurons make a form of memory where each neuron is able to remember its previous state. These properties of LSTMs make them a descent fit for the problem at hand. Implementation of the proposed network is detailed in section **Proposed Method Implementation**.

CHAPTER 5: EXPERIMENTS AND RESULTS

In this section we present implementation of baseline and proposed methods, their comparison and detailed study of importance of different features towards bank failure.

5.1 Evaluation Metrics

To evaluate the proposed algorithm and compare it with the base line implementations we chose accuracy, precision, recall and f1-Score as the performance measures. For this we generated a confusion matrix, table 7, where T_P = True positive (both actual and predicted class is survived), T_N = True negative (both actual and predicted class are failed), F_P = False positive (actual class is failed but predicted survived) and F_N = False negative (actual class is survived but predicted failed).

The description of these metrics is as follows:

Table 5-1: Confusion Matrix

Class		**Prediction**	
		Survived	**Failed**
Actual	**Survived**	T_P	F_N
	Failed	F_P	T_N

- **Accuracy**

  Accuracy is described as the ratio of correct predictions to the total number of predictions. The accuracy is calculated using equation **(5.1)**.

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \tag{5.1}$$

- **Precision**

  Precision is defined as the ratio of relevant predictions to the total number positive predictions. The precision is calculated using equation **(5.2)**.

$$Precision = \frac{T_P}{T_P + F_P} \tag{5.2}$$

- **Recall**

  Recall is the ratio of relevant predictions to the predictions that should have been if the algorithm was absolutely perfect. Recall is calculated using equation **(5.3)**.

$$Recall = \frac{T_P}{T_P + F_N} \tag{5.3}$$

- **F-Measure**

  Precision and Recall alone have their own importance but when combined using the harmonic mean give a more complete essence. F-Measure incorporates how many correct predictions are made and how many total predictions should have been made into a single value. The more closer these values are the more higher the score is. F-Measure is calculated using equation **(5.4)**.

$$FMeasure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5.4}$$

5.2 Baseline Implementations

We have chosen two of the most popular prediction methods: SVM and Logistic regression.

5.2.1 Support Vector Machines

The bank failure prediction is a binary classification problem and therefore SVM is a natural candidate for problem at hand. The features vectors with dimension $59 \times 1 \times \beta \mid \beta \in N$ are used to train single SVM, where N is the batch size hyper-parameter. Using (Chang & Lin, 2011) [44], the problem is formulated as a two-class soft-margin Support Vector Classification with regularization parameter C. The kernel is set to be the radial basis function (RBF) defined as follows (equation **(5.5)**):

$$K(x_i,x_j) = e^{-\gamma\,||x_i - x_i||^2} \tag{5.5}$$

The setting of parameters C and shape parameter γ is based on the combination that yields the best overall performance with a 10-fold cross-validation (Fushiki, 2011) [45] over training data. The results, reported in **Figure 5-1**, have been achieved by setting C as 1, γ as 0.25 and 'batch size' as 227. For further details of SVM we refer readers to (Chang & Lin, 2011) [44].

5.2.2 Logistic Regression

The second baseline method employed in our study is Logistic Regression (LR). LR maps one or more independent variables (59 bank features) to a dependent variable (probability of bank failure) shown in equation **(5.6)**.

$$p(x) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n)}} \tag{5.6}$$

Where $p(x)$ is the probability of bank failure, x_1, x_2 ... x_n are the input features and θ_0, θ_1 ... θ_n are the learning coefficients of the input features.

Different hyperparameters were used during the experimentation process with a 10-fold cross-validation (Fushiki, 2011) [45] over training data. Results with best set of hyperparameters have been, reported in **Figure 5-1**, with a 'learning rate' of 0.01, 15075 'training iterations' and Stochastic Gradient Descend optimizer. Softmax cross entropy loss function is used as our optimization objective function (equation **(5.7)**).

$$L = -\frac{1}{m}\sum_{i=1}^{m} \log\left(\frac{e^{f_{y_j}^{(i)}}}{\sum_{j=1}^{k} e^{f_j^{(i)}}}\right) \tag{5.7}$$

Where L is the loss, m is the total number of samples, k is the total number of output classes, y_j is the correct class for the i^{th} sample and $f(x^{(i)})$ is the mapping function for input feature vector $x^{(i)}$ and is described in equation 13. The LR implementation is based on Google's TensorFlow API (Abadi et al., 2016) [46].

5.3 Proposed Method Implementation

All the experiments were performed on a workstation with following specifications: Intel Core i7-3630QM 2.4 GHz, 4 GB Ram, NVIDIA GeForce GT-630M with 2GB DDR3 Memory and Windows 8.1. We have used Microsoft Visual studio 2017 v15.3 IDE along with python tools for visual studio (PTVS), python development language, and Google's TensorFlow API (Abadi et al., 2016) [46].

We implemented and trained the proposed 4-layer deep recurrent neural network comprising of LSTM cells. The network was designed to take a variable sequence length as input. The complete network diagram is shown in the **Figure 4-4**. Results with best set of hyper-parameters, reported in **Figure 5-1**, have been achieved with the following training parameters:

- Number of input features: 59
- Output classes: 2
- Maximum sequence length: 11
- Number of units in Cell: 128
- Network Layers: 4
- Optimizer: gradient descend optimizer
- Learning rate: 0.01
- Loss function: cross entropy loss
- Number of epochs: 225
- Batch size: 1
- Train set: 685 sequences
- test set 227 sequences
- Cross validation set 227 sequences
- Number of folds: 10

To prevent the neural network from over fitting we used k-Fold cross validation (Fushiki, 2011) [45] technique. The technique splits the data samples into three sets:

- train set, 60% of the dataset
- test set, 20% of the dataset
- cross validation set, 20% of the dataset

These sets are chosen randomly. A different random set is chosen in the next iteration. so that the whole data set could be fed to the network but not all at the same time. The network was trained on this 60% of data (train set) and the error was calculated on the test set. The results to be reported are taken on the cross-validation set. The average of the results of the k iterations is reported in figure 5. In our study we chose *k* as 10. The result comparison of the base line methods with proposed method (LSTMs) can be seen in **Figure 5-1**.

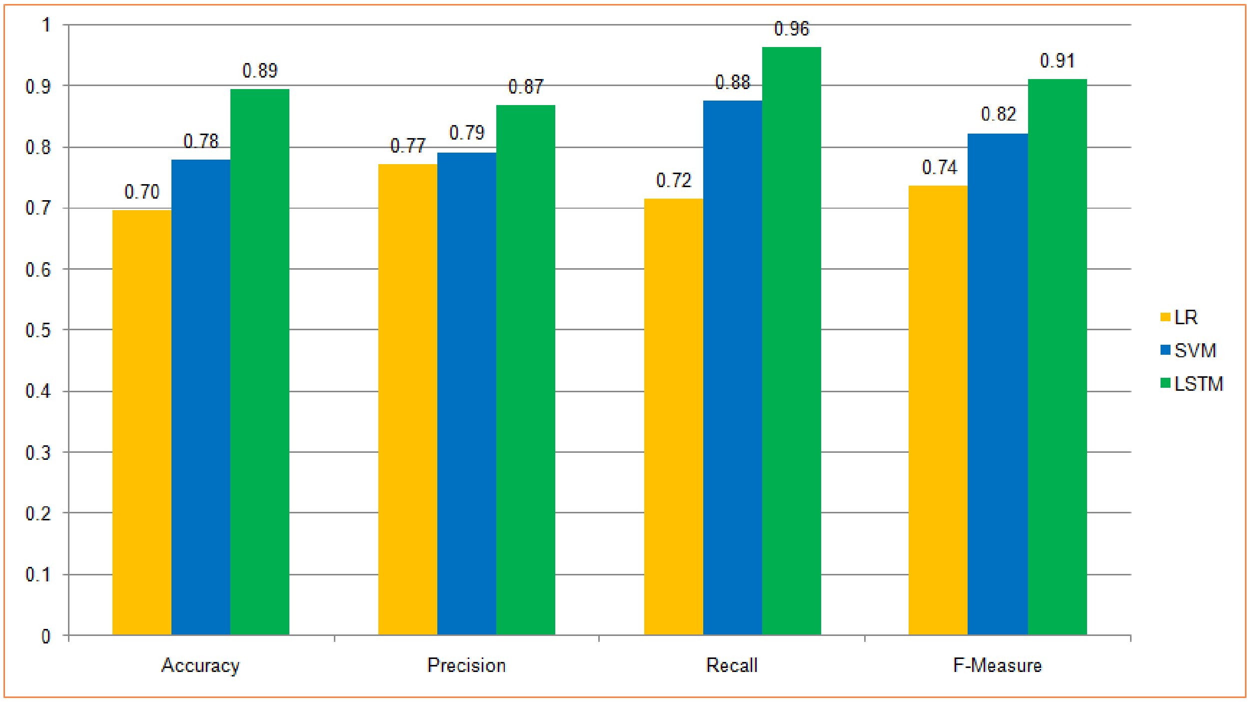

Figure 5-1: Comparison with Baseline Implementations

5.4 Ablation Study

Ablation study, in machine learning, is the study of the performance of an algorithm in which some of the data is hidden from it. It is done to test the algorithm for overfitting the problem. It also helps in the identification of key features in a complex problem. In our study we employed three different techniques to test the proposed algorithm.

5.4.1 Experiments with layers

Initially a three-layered LSTM network was trained. This basic network outperformed SVM and LR both. With the addition of another hidden layer (fourth layer) the network was able to perform even better than the three-layered network. A 3% increase in the F1-Score was observed.

Further experimentation was carried out in order to tune the 'batch size' hyperparameter. In these experiments batch size was set to 1139, 67, 17 and 1. We found out that setting it to 1 increased the F1-Score by 1.3%.

5.4.2 Feature importance experiment

The visualization of weights of an RNN is a complex task due to the masking of complex feature correlation and temporal variations. To get some insights we generated 10 different models for each cross-validation fold and performed the following experiments. The idea of these experiments was to generate an 'input sequence' that will be fed to these 10 trained models and their response will be observed collectively.

The concept of the first experiment (On Test) was to observe the change (increase in activations) in natural response of the network. In this experiment, an input sequence initialized with zeros and dimension 59×4 was given to the models. Any activation caused by this input will be the result of internal states of LSTM cells only. Once the natural responses of the models are known, the input is modified such that each CAMELS category is turned 'on' one by one, by setting their features to a higher value 1, in order to gauge their impact on the overall output of the models. The results of this experiment are shown in **Figure 5-2**.

Similarly, the concept of the second experiment (Off Test) was to observe the change (decrease in activations) in natural response of the network. In this experiment, an input sequence initialized with ones and dimension 59×4 was given to the models. Maximum activations took place due to this saturated input. Once the natural responses of the models are known, the input is modified such that each CAMELS category is turned 'off' one by one, by setting their features to a lower value 0, in order to gauge their impact on the overall output of the models. The experiment is opposite of the first one so the results of this are inverted and shown in figure **Figure 5-2**.

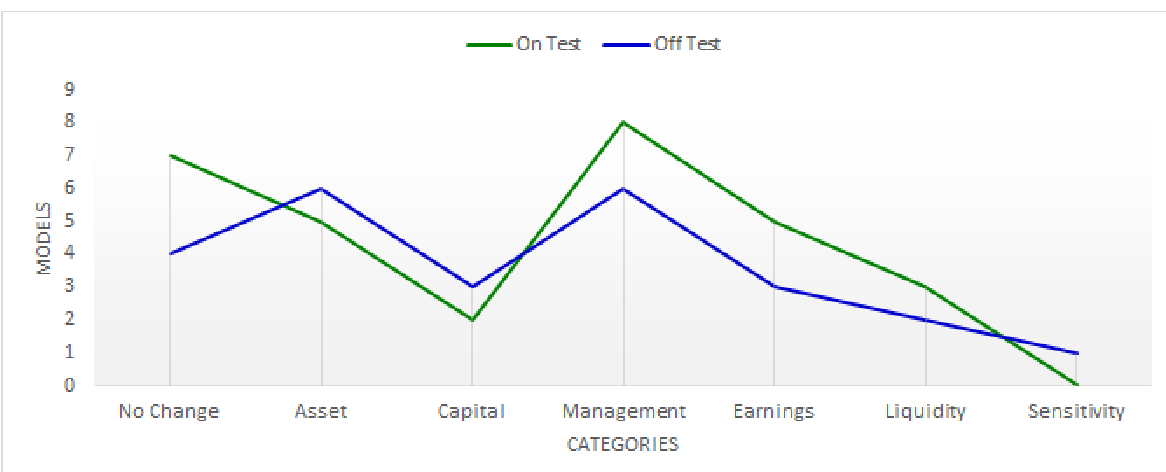

Figure 5-2: Feature Importance

The result show that the Management features had the highest impact on the survive-ability of the bank (figure 6 it can be seen that). The importance of the features is as follows:

$$Management > Asset > Earnings > Capital > Liquidity > Sensitivity$$

5.4.3 Sequence length experiment

Here we will test the solution for the second problem mentioned in **Challenges with the dataset**. **Figure 5-3** shows a joint frequency distribution of labels against the sequence lengths. Following relations between the sequence lengths and the output labels can be observed:

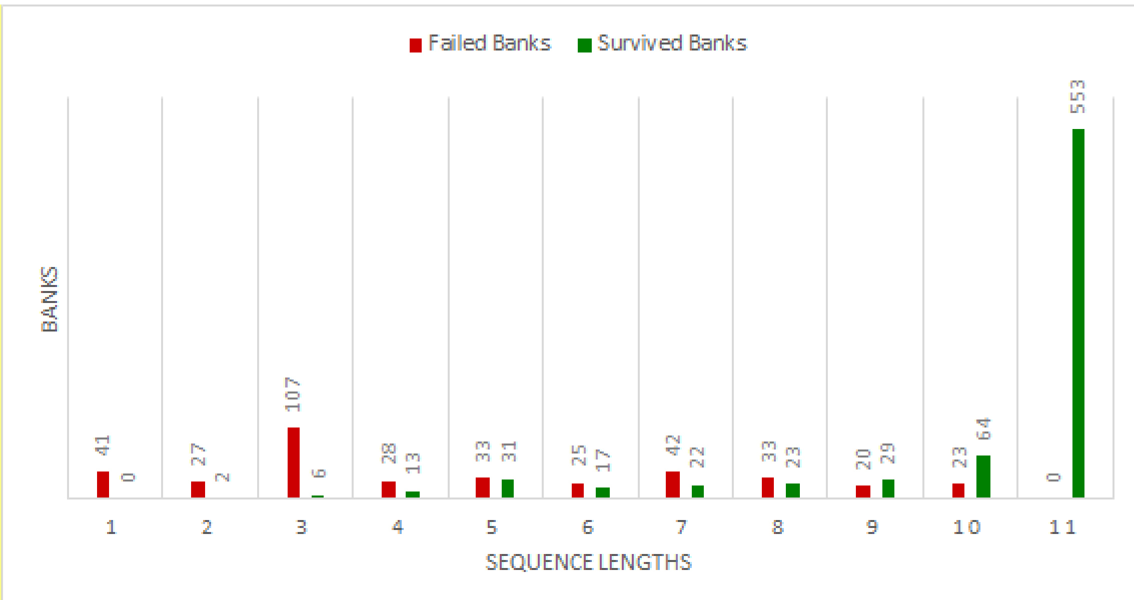

Figure 5-3: Sequence Lengths Comparison

These skewed patterns in the dataset can cause the network to get biased. It can cause the proposed method to over-fit on the given dataset by making it learn that the banks that have a sequence length greater than 10 can never fail, similarly banks that have a sequence length less than 4 will always fail, irrespective of the actual data.

To test this hypothesis sequences with $3 < lengths < 11$ were taken. This specific range contains samples from both classes, thus is the ideal candidate for evaluation. We found out that the network had 0.8694788 accuracy, 0.825224 precision, 0.940701 recall and 0.877459 F-Measure. We were expecting a slight decrease in performance as the correlation decreases when sequence lengths are reduced. The F1-score was reduced by a small amount (3.3%) proving that the model is not over-fit.

CHAPTER 6: Conclusion and Future work

The results showed that the LSTM can outperform Logistic regression and SVM by a considerable margin. The key reason behind these results is that Logistic regression and SVM treat the temporal data as feature vectors whereas LSTM keeps the temporal variation, of the data across the years, into account.

APPENDIX A

7.1 Detailed Results

The detailed results, against each cross-validation fold, average of the cross-validation models, Multi-layer experimentation and with different batch sizes are given below:

7.1.1 Loss

Algorithm	K=0	K=1	K=2	K=3	k=4	k=5	k=6	k=7	k=8	k=9	Average
SVM (Baseline method 1)											#DIV/0!
Logistic Regression (Baseline method 2)	0.651745	0.59419	0.556635	0.543241	0.547708	0.614933	0.664622	0.73822	0.765323	0.624908	0.6301525
LSTM (Proposed Method)	0.017122	0.069494	0.042781	0.053886	0.026697	0.034179	0.046162	0.051408	0.151963	0.010428	0.050412
LSTM 2 Layer Batch Size 17 (Proposed Method)	0.00056	0.038062	0.000739	0.000855	0.011685	0.00266	0.029982	0.016259	0.050833	0.000215	0.015185
LSTM 2 Layer Batch Size 1 (Proposed Method)	0.057314	0.095536	0.052382	0.136911	0.046456	0.056103	0.167135	0.17765	0.627328	0.006044	0.1422859

7.1.2 Accuracy

Algorithm	K=0	K=1	K=2	K=3	k=4	k=5	k=6	k=7	k=8	k=9	Average
SVM (Baseline method 1)											#DIV/0!
Logistic Regression (Baseline method 2)	0.651745	0.59419	0.556635	0.543241	0.547708	0.614933	0.664622	0.73822	0.765323	0.624908	0.6301525
LSTM (Proposed Method)	0.017122	0.069494	0.042781	0.053886	0.026697	0.034179	0.046162	0.051408	0.151963	0.010428	0.050412
LSTM 2 Layer Batch Size 17 (Proposed Method)	0.00056	0.038062	0.000739	0.000855	0.011685	0.00266	0.029982	0.016259	0.050833	0.000215	0.015185
LSTM 2 Layer Batch Size 1 (Proposed Method)	0.057314	0.095536	0.052382	0.136911	0.046456	0.056103	0.167135	0.17765	0.627328	0.006044	0.1422859

7.1.3 Precision

Algorithm	K=0	K=1	K=2	K=3	k=4	k=5	k=6	k=7	k=8	k=9	Average
SVM (Baseline method 1)	0.96243	0.96036	0.89895	0.87436	0.8373	0.84703	0.83063	0.56618	0.66577	0.46828	0.791129
Logistic Regression (Baseline method 2)	0.95418	0.95974	0.88942	0.86972	0.84458	0.84333	0.83137	0.49288	0.56836	0.4576	0.771118

LSTM (Proposed Method)	0.97895	0.98	0.92683	0.91566	0.92208	0.94805	0.92063	0.67857	0.7963	0.51429	0.858136
LSTM 2 Layer Batch Size 17 (Proposed Method)	1	0.98058	0.95402	0.94118	0.95122	0.97333	0.92063	0.73585	0.89362	0.61667	0.89671
LSTM 2 Layer Batch Size 1 (Proposed Method)	1	0.97087	0.92784	0.93182	0.88764	0.87912	0.8875	0.78	0.78689	0.63077	0.868245

7.1.4 Recall

Algorithm	K=0	K=1	K=2	K=3	k=4	k=5	k=6	k=7	k=8	k=9	Average
SVM (Baseline method 1)	0.92844	0.83247	0.81025	0.91586	0.93054	0.91564	0.86161	0.81053	0.86667	0.88261	0.875462
Logistic Regression (Baseline method 2)	0.66354	0.76364	0.71579	0.80195	0.77716	0.8625	0.77091	0.58562	0.51872	0.69401	0.715384
LSTM (Proposed Method)	0.95876	0.93333	0.8	0.90476	0.86585	0.9125	0.77333	0.88372	0.84314	0.87805	0.875344
LSTM 2 Layer Batch Size 17 (Proposed Method)	0.98969	0.9619	0.87368	0.95238	0.95122	0.9125	0.77333	0.90698	0.82353	0.90244	0.904765
LSTM 2 Layer Batch Size 1 (Proposed Method)	1	0.95238	0.94737	0.97619	0.96341	1	0.94667	0.90698	0.94118	1	0.963418

7.1.5 F-Measure

Algorithm	K=0	K=1	K=2	K=3	k=4	k=5	k=6	k=7	k=8	k=9	Average
SVM (Baseline method 1)	0.94513	0.89185	0.8523	0.89463	0.88146	0.88	0.84584	0.66667	0.75305	0.61191	0.822284
Logistic Regression (Baseline method 2)	0.78275	0.85053	0.79321	0.83446	0.80947	0.85281	0.8	0.53527	0.5424	0.55154	0.735244
LSTM (Proposed Method)	0.96875	0.9561	0.85876	0.91018	0.89308	0.92994	0.84058	0.76768	0.81905	0.64865	0.859277
LSTM 2 Layer Batch Size 17 (Proposed Method)	0.99482	0.97115	0.91209	0.94675	0.95122	0.94194	0.84058	0.8125	0.85714	0.73267	0.896086
LSTM 2 Layer Batch Size 1 (Proposed Method)	1	0.96154	0.9375	0.95349	0.92698	0.93567	0.91613	0.83871	0.85714	0.77358	0.910074

REFERENCES

[1] E. I. Altman, "Financial ratios, discriminant analysis and the prediction of corporate bankruptcy," *The journal of finance,* vol. 23, pp. 589-609, 1968.

[2] G. V. Karels and A. J. Prakash, "Multivariate normality and forecasting of business bankruptcy," *Journal of Business Finance & Accounting*, vol. 14, pp. 573-593, 1987.

[3] J. A. Haslem, C. A. Scheraga and J. P. Bedingfield, "An analysis of the foreign and domestic balance sheet strategies of the US banks and their association to profitability performance," *MIR: Management International Review,* pp. 55-75, 1992.

[4] R. C. West, "A factor-analytic approach to bank condition," *Journal of Banking & Finance,* vol. 9, pp. 253-266, 1985.

[5] E. P. Davis and D. Karim, "Comparing early warning systems for banking crises," *Journal of Financial stability,* vol. 4, pp. 89-120, 2008a.

[6] E. P. Davis and D. Karim, "Could early warning systems have helped to predict the sub-prime crisis?," *National Institute Economic Review,* vol. 206, pp. 35-47, 2008b.

[7] S. Canbas, A. Cabuk and S. B. Kilic, "Prediction of commercial bank failure via multivariate statistical analysis of financial structures: The Turkish case," *European Journal of Operational Research,* vol. 166, pp. 528-546, 2005.

[8] A. E. Celik and Y. Karatepe, "Evaluating and forecasting banking crises through neural network models: An application for Turkish banking sector," *Expert systems with Applications,* vol. 33, pp. 809-815, 2007.

[9] P. Alam, D. Booth, K. Lee and T. Thordarson, "The use of fuzzy clustering algorithm and self-organizing neural networks for identifying potentially failing banks: an experimental study," *Expert Systems with Applications,* vol. 18, pp. 185-199, 2000.

[10] M. A. Boyacioglu, Y. Kara and Ö. K. Baykan, "Predicting bank financial failures using neural networks, support vector machines and multivariate statistical methods: A comparative analysis in the sample of savings deposit insurance fund (SDIF) transferred banks in Turkey," *Expert Systems with Applications,* vol. 36, pp. 3355-3366, 2009.

[11] K. Y. Tam, "Neural network models and the prediction of bank bankruptcy," *Omega,* vol. 19, pp. 429-445, 1991.

[12] K. Y. Tam and M. Y. Kiang, "Managerial applications of neural networks: the case of bank failure predictions," *Management science,* vol. 38, pp. 926-947, 1992.

[13] T. B. Bell, "Neural nets or the logit model? A comparison of each model's ability to predict commercial bank failures," *Intelligent Systems in Accounting, Finance & Management,* vol. 6, pp. 249-264, 1997.

[14] P. Swicegood and J. A. Clark, "Off-site monitoring systems for predicting bank underperformance: a comparison of neural networks, discriminant analysis, and professional human judgment," *Intelligent Systems in Accounting, Finance & Management,* vol. 10, pp. 169-186, 2001.

[15] J. Kolari, D. Glennon, H. Shin and M. Caputo, "Predicting large US commercial bank failures," *Journal of Economics and Business,* vol. 54, pp. 361-387, 2002.

[16] G. Lanine and R. Vander Vennet, "Failure prediction in the Russian bank sector with logit and trait recognition models," *Expert Systems with Applications,* vol. 30, pp. 463-478, 2006.

[17] M. L. Marais, J. M. Patell and M. A. Wolfson, "The experimental design of classification models: An application of recursive partitioning and bootstrapping to commercial bank loan classifications," *Journal of accounting Research,* pp. 87-114, 1984.

[18] H. Frydman, E. I. Altman and D.-L. Kao, "Introducing recursive partitioning for financial classification: the case of financial distress," *The Journal of Finance,* vol. 40, pp. 269-291, 1985.

[19] Z. Pawlak, "Rough sets," *International journal of computer & information sciences,* vol. 11, pp. 341-356, 1982.

[20] S. Greco, B. Matarazzo and R. Slowinski, "A new rough set approach to multicriteria and multiattribute classification," in *International Conference on Rough Sets and Current Trends in Computing*, 1998.

[21] B. S. Ahn, S. S. Cho and C. Y. Kim, "The integrated methodology of rough set theory and artificial neural network for business failure prediction," *Expert systems with applications,* vol. 18, pp. 65-74, 2000.

[22] H. Zhao, A. P. Sinha and W. Ge, "Effects of feature construction on classification performance: An empirical study in bank failure prediction," *Expert Systems with Applications,* vol. 36, pp. 2633-2644, 2009.

[23] V. Vapnik, The nature of statistical learning theory, Springer science & business media, 2013.

[24] W.-H. Chen and J.-Y. Shih, "A study of Taiwan's issuer credit rating systems using support vector machines," *Expert Systems with Applications,* vol. 30, pp. 427-435, 2006.

[25] Z. Huang, H. Chen, C.-J. Hsu, W.-H. Chen and S. Wu, "Credit rating analysis with support vector machines and neural networks: a market comparative study," *Decision support systems,* vol. 37, pp. 543-558, 2004.

[26] K.-S. Shin, T. S. Lee and H.-j. Kim, "An application of support vector machines in bankruptcy prediction model," *Expert Systems with Applications,* vol. 28, pp. 127-135, 2005.

[27] X. Luo, "Evaluating the profitability and marketability efficiency of large banks: An application of data envelopment analysis," *Journal of Business research,* vol. 56, pp. 627-635, 2003.

[28] N. K. Avkiran, "Opening the black box of efficiency analysis: an illustration with UAE banks," *Omega,* vol. 37, pp. 930-941, 2009.

[29] C. Kao and S.-T. Liu, "Predicting bank performance with financial forecasts: A case of Taiwan commercial banks," *Journal of Banking & Finance,* vol. 28, pp. 2353-2368, 2004.

[30] E. G. Tsionas and E. N. Papadakis, "A Bayesian approach to statistical inference in stochastic DEA," *Omega,* vol. 38, pp. 309-314, 2010.

[31] K. Kosmidou and C. Zopounidis, "Predicting US commercial bank failures via a multicriteria approach," *International Journal of Risk Assessment and Management,* vol. 9, pp. 26-43, 2008.

[32] C. Gaganis, F. Pasiouras and C. Zopounidis, "A multicriteria decision framework for measuring banks' soundness around the world," *Journal of Multi-Criteria Decision Analysis,* vol. 14, pp. 103-111, 2006.

[33] F. Pasiouras, C. Gaganis and M. Doumpos, "A multicriteria discrimination approach for the

credit rating of Asian banks," *Annals of Finance,* vol. 3, pp. 351-367, 2007.

[34] M. P. Niemira and T. L. Saaty, "An analytic network process model for financial-crisis forecasting," *International Journal of Forecasting,* vol. 20, pp. 573-587, 2004.

[35] V. Ravi and C. Pramodh, "Threshold accepting trained principal component neural network and feature subset selection: Application to bankruptcy prediction in banks," *Applied Soft Computing,* vol. 8, pp. 1539-1548, 2008.

[36] B. Back, T. Laitinen and K. Sere, "Neural networks and genetic algorithms for bankruptcy predictions," *Expert Systems with Applications,* vol. 11, pp. 407-413, 1996.

[37] H. Jo and I. Han, "Integration of case-based forecasting, neural network, and discriminant analysis for bankruptcy prediction," *Expert Systems with applications,* vol. 11, pp. 415-422, 1996.

[38] W. L. Tung, C. Quek and P. Cheng, "GenSo-EWS: a novel neural-fuzzy based early warning system for predicting bank failures," *Neural networks,* vol. 17, pp. 567-587, 2004.

[39] G. S. Ng, C. Quek and H. Jiang, "FCMAC-EWS: A bank failure early warning system based on a novel localized pattern learning and semantically associative fuzzy neural network," *Expert Systems with Applications,* vol. 34, pp. 989-1003, 2008.

[40] A. Cielen, L. Peeters and K. Vanhoof, "Bankruptcy prediction using a data envelopment analysis," *European Journal of Operational Research,* vol. 154, pp. 526-532, 2004.

[41] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems,* vol. 6, pp. 107-116, 1998.

[42] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation,* vol. 9, pp. 1735-1780, 1997.

[43] J. Chung, C. Gulcehre, K. Cho and Y. Bengio, "Gated feedback recurrent neural networks," in *International Conference on Machine Learning*, 2015.

[44] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM transactions on intelligent systems and technology (TIST),* vol. 2, p. 27, 2011.

[45] T. Fushiki, "Estimation of prediction error by using K-fold cross-validation," *Statistics and Computing,* vol. 21, pp. 137-146, 2011.

[46] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard and others, "Tensorflow: a system for large-scale machine learning.," in *OSDI,* 2016.

[47] C. Zopounidis and M. Doumpos, "A multicriteria decision aid methodology for sorting decision problems: The case of financial distress," *Computational Economics,* vol. 14, pp. 197-218, 1999.

[48] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146,* 2016.

[49] J. B. Taylor, "The financial crisis and the policy responses: An empirical analysis of what went wrong," 2009.

[50] R. K. Srivastava, K. Greff and J. Schmidhuber, "Highway networks," *arXiv preprint arXiv:1505.00387,* 2015.

[51] R. Pascanu, T. Mikolov and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International Conference on Machine Learning,* 2013.

[52] I. Olmeda and E. Fernández, "Hybrid classifiers for financial multicriteria decision making: The case of bankruptcy prediction," *Computational Economics,* vol. 10, pp. 317-335, 1997.

[53] P. R. Kumar and V. Ravi, "Bankruptcy prediction in banks and firms via statistical and intelligent techniques--A review," *European journal of operational research,* vol. 180, pp. 1-28, 2007.

[54] M. D. Fethi and F. Pasiouras, "Assessing Bank Performance with Operational Research and Artificial Intelligence Techniques: A Survey, University of Bath, School of Management," 2009.

[55] M. Doumpos and C. Zopounidis, "Assessing financial risks using a multicriteria sorting procedure: the case of country risk assessment," *Omega,* vol. 29, pp. 97-109, 2001.

[56] G. Dell'Ariccia, D. Igan and L. Laeven, Credit booms and lending standards: Evidence from the subprime mortgage market, International Monetary Fund, 2008.

[57] F. F. I. E. Council, "Uniform financial institutions rating system," *Federal Register,* vol. 61, pp. 67021-9, 1996.

[58] M. Brummermeier, "Deciphering the Liquidity and Credit Crunch 2007-08," *Journal of*

Economic Perspectives, vol. 23, pp. 77-100, 2009.