# Prediction of Corporate Finance Incidents Using Machine Learning Algorithms

Sheetal Rani Prasad, Siva Preethi Ramesh
San Diego State University

## Abstract

Financial crisis prediction is the most challenging and anticipated issue to be tackled in corporate firms, small businesses, investors, and even governments. When a firm is acquired, it signifies that another company has bought it in order to gain control of it and merge it with another company. As a result of the transformation, firm stakeholders are now able to make business decisions that might help the broader organization achieve its objectives or completely absorb the firm into another. This report evaluates various machine learning algorithms that will help to predict whether a company is going to get acquired or not based on the financial data available for that firm.

## Introduction

Financial asset values are non-linear, volatile, and chaotic, making them challenging to predict financial time series. For financial institutions, the ability to predict or forecast business failures is crucial, as incorrect decisions can have direct financial consequences. Machine learning models are among the most investigated among the most recent methodologies, thanks to their ability to recognize complicated patterns in a variety of applications.

Given the advantages that machine learning brings to finance, we focused on applying supervised machine learning to a financial incident problem. The project works with various algorithms to predict the possible outcome that a firm might get acquired or not. The algorithm works on the revenue and capital data of company.

The rest of the present paper is organized as follows. Section 2 explains the data set available; Section 3 goes through some basic concepts. Finally, section 4 describes the details of the financial crisis prediction and the observations made during this project work.

## Dataset

The dataset contains features such as fiscal year, total assets of the firm, total revenue of the firm, net profit of the firm, current assets of the firm, capital expenditure of the firm, shareholder’s equity and so on. The project uses supervised algorithms; therefore, the data contains a column named target in which the value 0 represents not acquired and value 1 represents acquired.

For the sequential models, we use features from the previous years to predict whether the same company will be acquired in the next year.

We can use at most the previous 5 years to predict the next year, so for the year 2001, there is only one historical data point. But for the year 2010, there can be 5 data points that can be used for the prediction.

We process and clean the data as required for the sequential models.

# Methods

3.1 SVM

The support vector machine algorithm's goal is to find a hyperplane in an Ndimensional space that distinguishes between data points. The SVM principle is fairly intuitive and simple to grasp. SVM can be used to build numerous separating hyperplanes from labelled data, dividing the data space into segments with only one type of data in each segment. The SVM technique is often useful for data with nonregularity, or data with an uncertain distribution.

3.2 Logistic Regression Logistic Regression is a Machine Learning algorithm that is used to solve classification problems. It is a predictive analytic approach that is based on the probability notion. The Sigmoid method transforms any real number between 0 and 1 into a number between 0 and 1. We utilize sigmoid to map predictions to probabilities in machine learning.

3.3 Decision Tree

Decision Trees are a type of Supervised Machine Learning in which data is continually separated based on a parameter. Two entities, decision nodes and leaves, can be used to explain the tree. The decisions or final outcomes are represented by the leaves. And the data is separated at the decision nodes.

3.4 MLP Classifier

A feedforward artificial neural network model, the multilayer perceptron (MLP), maps input data sets to a collection of corresponding outputs. An MLP is composed of numerous layers, each of which is fully connected to the one before it. Except for the nodes of the input layer, the nodes of the layers are neurons with nonlinear activation functions. One or more nonlinear hidden layers may exist between the input and output layers.

3.5 Recurrent Neural Network (RNN)

A recurrent neural network (RNN) is a type of artificial neural network designed to recognize data's sequential patterns to predict the following scenarios. This architecture is especially powerful because of its nodes connections, allowing the exhibition of a temporal dynamic behaviour. The use of feedback loops to process a sequence is another important feature.

Here we valuate two different architectures called the Long Short-Term Memory (LSTM) and the Gated Recurrent Units (GRU).

A many-to-one recurrent architecture can be used for this task. Using t1, t2, t3, t4 to denote the years from 2006 to 2009. For each year, we can use its column E to R as a feature vector fi, (i=1,2,3,4), and feed them to an RNN model, and use the model's outputs to predict the binary variable: whether the company got acquired.

A common LSTM unit is composed of a cell, an input gate, an output gate[13] and a forget gate.[14] The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.

The GRU is like a long short-term memory (LSTM) with a forget gate. We define our LSTM and GRU modules by extending PyTorch's nn.Module which is a base class for all neural network modules.

## Observations

### 4.1 SVM

The SVM has performed poorly on the given dataset. The confusion matrix of the model shows that the dataset is skewed, containing about more than 98% not acquired data and only 2% acquired data. This results in poor precision from the model.

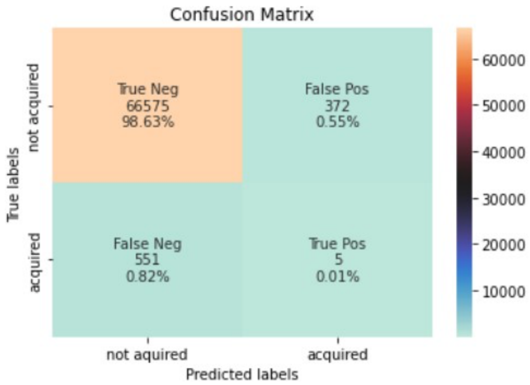


Fig. Confusion Matrix for SVM

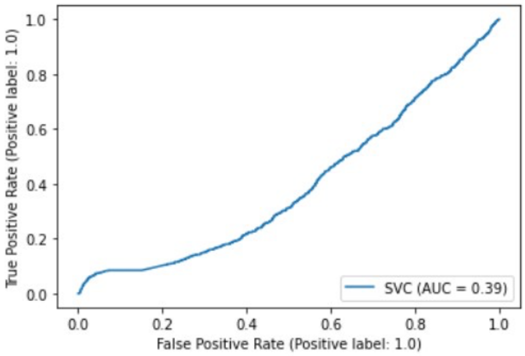


Fig. ROC curve for SVM

### 4.2 Logistic Regression

Logistic Regression performs a little better compared to SVM with accuracy of 99.16%. Again, the result is impacted by the majority of not acquired data in the test set.

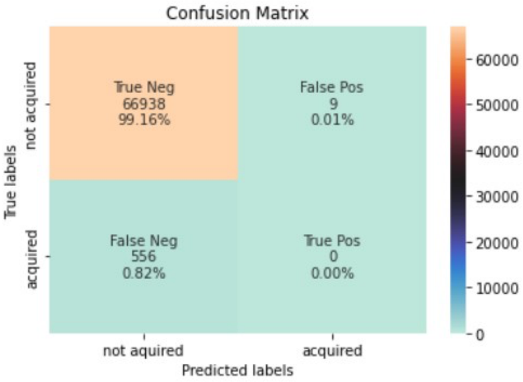


Fig. Confusion Matrix for Logistic Regression

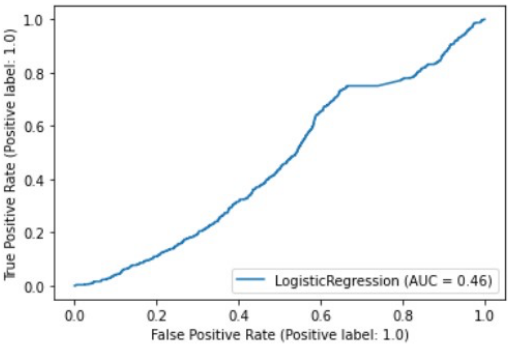


Fig. ROC curve for Logistic Regression

### 4.3 Decision Tree

The decision tree improves the classification of the positive values, where we can see that the TP values jumped from the value in Logistic Regression model. The accuracy is dropped to 98.50% compared to the same logistic regression.

Even this model verifies that the data set is imbalanced towards the ‘not acquired’ value.

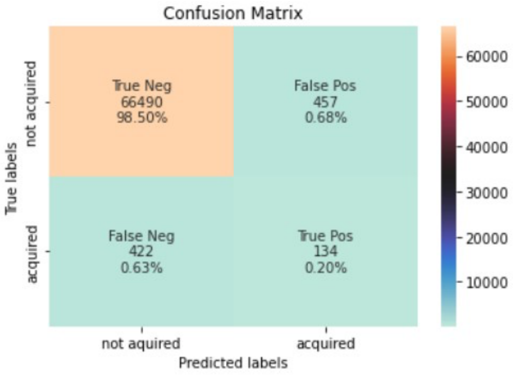


Fig. Confusion Matrix for Decision Tree

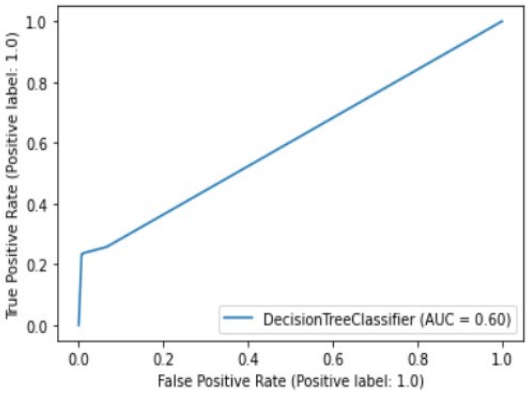


Fig. ROC Curve for Decision Tree

### 4.4 MLP Classifier

This last non-sequential model has the best accuracy and F1-Score compared to the rest of the non-sequential model. The prediction of True positive is till poor as a result of imbalanced dataset. However, the prediction of 'not acquired' that is True Negative is 99.17%.

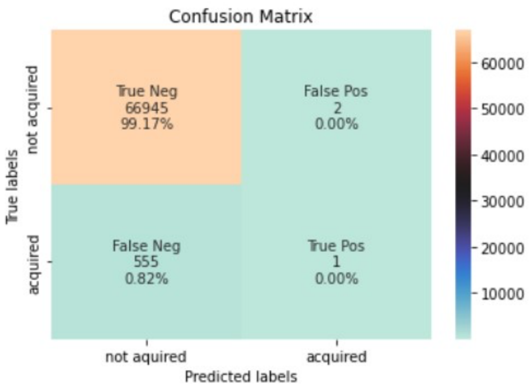


Fig. Confusion Matrix for MLP Classifier

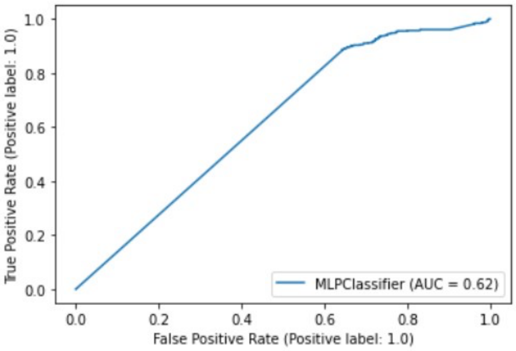
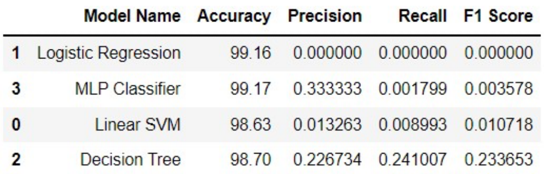


Fig. ROC Curve for MLP Classifier

The F1-score uses the harmonic mean of a classifier's precision and recall to create a single statistic. The F1 score aims to combine the precision and recall measurements into a single value. The F1 score was created with unbalanced data in mind.

When True Positives and True Negatives are more significant, Accuracy is employed, but F1-score is used when False Negatives and False Positives are critical.

When the class distribution is similar, accuracy can be employed, but F1-score is a better statistic when there are imbalanced classes, as is the case of this project. Here is the table sorted based on the F1-Score for the non-sequential models of the project.



| | Model Name | Accuracy | Precision | Recall | F1 Score |
|---|---|---|---|---|---|
| 1 | Logistic Regression | 99.16 | 0.000000 | 0.000000 | 0.000000 |
| 3 | MLP Classifier | 99.17 | 0.333333 | 0.001799 | 0.003578 |
| 0 | Linear SVM | 98.63 | 0.013263 | 0.008993 | 0.010718 |
| 2 | Decision Tree | 98.70 | 0.226734 | 0.241007 | 0.233653 |

Table: All models with all metrics

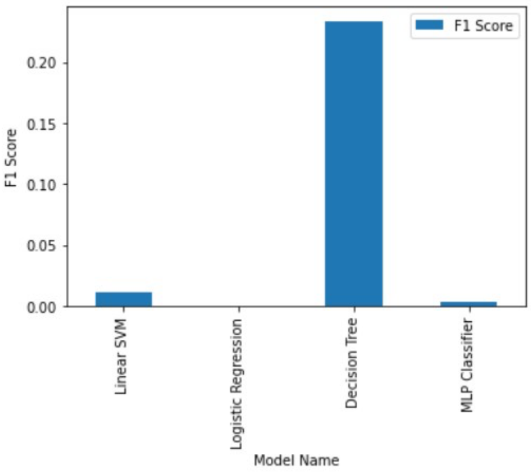


Fig: Bar graph for the F1-Score of all nonsequential model

4.5 Sequential Models

Different hyperparameters for the LSTM and the GRU Sequential Models were experimented with and studied including different learning rate in the optimization algorithms, the number of hidden layers in the neural network and the number of epochs.

The learning rate was set at different values for both the LSTM and GRU models to understand their behaviour. If the learning rate is set too low, training progressed very slowly as we are making very tiny updates to the weights in your network.

The GRU model training was much faster compared to the LSTM model in terms of speed for all learning rates.

## Code Repository

All data and code can be accessed on Github at https://github.com/sheetalrprasad/CS549_Grp13_Fin

The setup instructions are provided in the README.md file in the repository.

## Author Contribution

The non-sequential algorithms were implemented by Sheetal Rani Prasad and the Sequential algorithm implementation and experiments were done by Sivapreethi Ramesh. The observations recorded in this report were run using python with sklearn, pytorch libraries on Jupyter notebook.

## References

[1] W. Lin, Y. Hu and C. Tsai, "Machine Learning in Financial Crisis Prediction: A Survey," in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 42, no. 4, pp. 421-436, July 2012, doi: 10.1109/TSMCC.2011.2170420.

[2] Stock Price Prediction with PyTorch, LSTM and GRU to predict Amazon’s stock prices https://medium.com/swlh/stock-price-prediction-with-pytorch-37f52ae84632

[3] A PyTorch Example to Use RNN for Financial Prediction. https://chandlerzuo.github.io/blog/2017/11/darnn