Big Data Analytics towards Guaranteed Loan repayment of Corporate Firms

Shilpa.S.Ghasti
CSE 4th Semester ,CMRIT
shilpasghasti@gmail.com

Swathi.Y
Associate Professor & HOD,CMRIT
swathi.y@cmrit.ac.in

Abstract

The loan repayment prediction models are needed by financiers like banks in order to check the repayment of loans from companies. A very strong model needs a very large amount of data with periodic updates. Such size of data cannot be processed straightforwardly by the tools used in building Bankruptcy Prediction Models; however Big Data Analytics offers the chance to analyze such data.We propose firstly, Data Collection & Pre-processing then Secondly instead of using single classifier, we will use a Ensemble based approach where multiple classifiers are used and the results are combined to arrive at prediction result. This way classifier accuracy can be improved.

Keywords: Bankrupty Prediction,Data Collection and Preprocessing, Classifier,Loan Repayment

I. Introduction

The intensity of business failures in the likes of the construction and manufacturing industries has led to continuous advancement in bankruptcy prediction research, usually in the form of developing bankruptcy prediction models (BPMs) using various tools such as artificial neural networks, rough set, etc. The use of BPMs to identify potential construction business failure can help prevent failures as well as ensure credit or contracts are given only to healthy construction companies.

II. Related Work

[1] Jae H. Min , Young-Chan Lee proposed Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters

They have focussed mainly on Applied SVM to bankruptcy prediction problem. SVM transforms complex problems into simpler problems that can use linear discriminated functions.

[2] A.Martin,M.Manjula and Dr.Prasanna Venkatesan proposed A Business Intelligence Model to Predict Bankruptcy using Financial Domain Ontology with Association Rule Mining Algorithm.

Developed an ontological model called Altman Z-score model. Uses financial rations to predict bankruptcy.

$$Z= 1.2X_1 + 1.4X_2 + 3.3X_3 + 0.6X_4 +0.999X_5$$

$Z > 2.99$ -“Safe” Zone

$1.81 < Z < 2.99$ -“Gray” Zone

$Z < 1.81$ -“Distress” Zone

[3] Maria Reznakova, Michal Karas proposed Bankruptcy Prediction Models: Can the prediction power of the models be improved by using dynamic indicators?

Aims to analyze the potential dynamic financial ratios. A non parametric boosted tree method was used .It Captured the complex relationship between the predictors.

III. Existing System

It focuses on developing a framework on how Big Data and Machine Learning (ML) can be used to build an updatable and append-able robust failure prediction system for the construction industry of any country with large records of data. The following diagram shows the existing system as explained above. Here only one classifier is used.

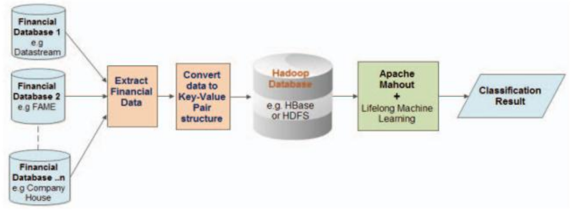

Financial Database 1 e.g Datastream
Financial Database 2 e.g FAME
Financial Database ..n e.g Company House
Extract Financial Data
Convert data to Key-Value Pair structure
Hadoop Database
e.g. HBase or HDFS
Apache Mahout + Lifelong Machine Learning
Classification Result

Fig 1: Flow diagram for application of Big Data Analytics on bankruptcy prediction

In the framework, huge financial data of construction companies is extracted from numerous data sources and converted to the Key-Value Pair structure before being imported into a Big Data Analytics database such as HBase. Apache Mahout with LML is subsequently used to perform a classification analysis on the huge data using a BPM machine learning tool. This produces a classification result which predicts firms as either failing or non-failing. The LML is then employed for training every time new data is appended in order to avoid the intensive retraining u sing the full data.

IV. Proposed System

The following changes takes place in the proposed system.

1. **<u>Data Collection & Pre-processing</u>**: We will collect data from various financial sources like DataStream and calculate the entropy of features to the classification result and from it choose the best features for classification.
2. **<u>Classifier</u>**: Instead of using single classifier, we will use a Ensemble based approach where multiple classifiers are used and the results are combined to arrive at prediction result. This way classifier accuracy can be improved.

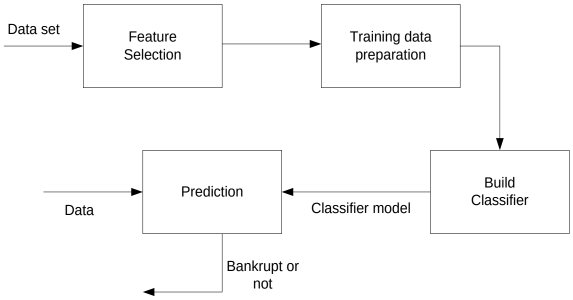


Fig 2. System architecture of proposed system

The modules shown in above diagram are as follows

1.Feature selection: This module takes input in the form of data set. Data set may contain financial statements for the companies or data from many sources.

In machine learning ,statistics feature selection is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. Feature selection techniques are used for three reasons: Simplification of models to make them easier to interpret by researchers/users, Shorter training times and enhanced generalization by reducing over fitting

The central premise when using a feature selection technique is that the data contains many features that are either redundant or irrelevant, and can thus be removed without incurring much loss of information.

2.Training data preparation: Training data set is created in this phase. Before build classifier model training data set has to be created. Data preparation is a pre-processing step in which data from one or more sources is cleaned and transformed to improve its quality prior to its use.The procedure is explained as in the figure below.

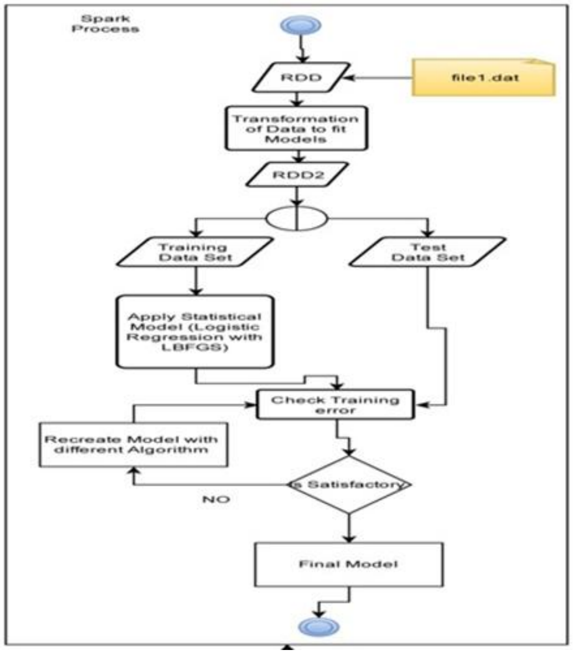

Spark Process
RDD
file1.dat
Transformation of Data to fit Models
RDD2
Training Data Set
Test Data Set
Apply Statistical Model (Logistic Regression with LBFGS)
Check Training error
Recreate Model with different Algorithm
s Satisfactory
NO
Final Model

Fig 3: Classification of Data Sets

Following classifiers of SPARK ML Library are used for classification.

Logistic Regression

Logistic regression measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function, which is the cumulative logistic distribution.

Logistic regression can be seen as a special case of generalized linear model and thus analogous to linear regression. The model of logistic regression, however, is based on quite different assumptions from those of linear regression. In particular the key differences of these two models can be seen in the following two features of logistic regression. First, the conditional distribution $y \mid x$ is a Bernoulli distribution rather than a Gaussian distribution, because the dependent variable is binary. Second, the predicted values are probabilities and are therefore restricted to (0,1) through the logistic distribution function because logistic regression predicts the probability of particular outcomes.

Logistic Regression With L – BFGS

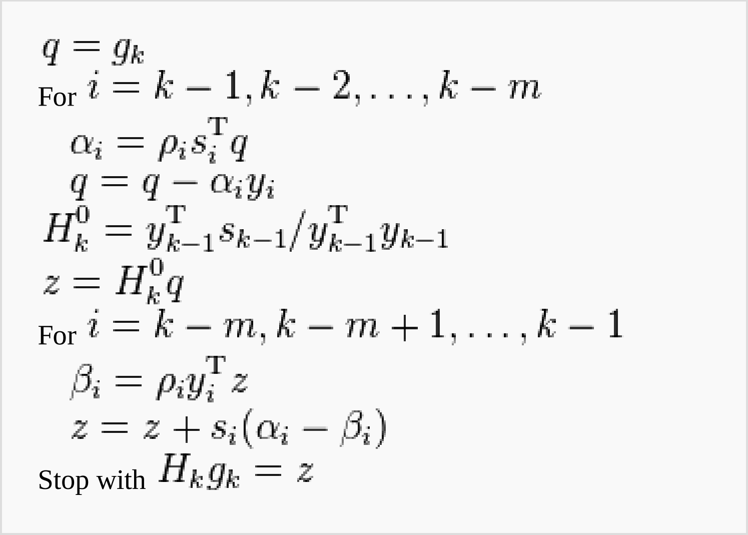
Limited-memory BFGS (L-BFGS or LM-BFGS) is an optimization algorithm in the family of quasi-Newton methods that approximates the Broyden–Fletcher–Goldfarb–Shanno(BFGS) algorithm using a limited amount of computer memory. It is a popular algorithm for parameter estimation in machine learning.

Like the original BFGS, L-BFGS uses estimation to the inverse Hessian matrix to steer its search through variable space, but where BFGS stores a dense *n*×*n* approximation to the inverse Hessian (*n* being the number of variables in the problem), L-BFGS stores only a few vectors that represent the approximation implicitly. Due to its resulting linear memory requirement, the L-BFGS method is particularly well suited for optimization problems with a large number of variables. Instead of the inverse Hessian $\mathbf{H}_k$, L-BFGS maintains a history of the past *m* updates of the position **x** and gradient ∇*f*(**x**), where generally the history size *m* can be small (often *m*<10). These updates are used to implicitly do operations requiring the $\mathbf{H}_k$-vector product.

Algorithm:

L-BFGS shares many features with other quasi-Newton algorithms, but is very different in how the matrix-vector multiplication for finding the search direction is carried out $d_k = -H_k g_k$. There are multiple published approaches using a history of updates to form this direction vector. Here, we give a common approach, the so-called "two loop recursion. We'll take as given x_k, the position at the k-th iteration, and $g_k \equiv \nabla f(x_k)$ where f is the function being minimized, and all vectors are column vectors. We also assume that we have stored the last m updates of the form $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$.

We'll define $\rho_k = \frac{1}{y_k^{\mathrm{T}} s_k}$, and H_k^0 will be the 'initial' approximate of the inverse Hessian that our estimate at iteration k begins with. Then we can compute the (uphill) direction as shown in figure 4.



```
$q = g_k$
For $i = k-1, k-2, \ldots, k-m$
  $\alpha_i = \rho_i s_i^{\mathrm{T}} q$
  $q = q - \alpha_i y_i$
$H_k^0 = y_{k-1}^{\mathrm{T}} s_{k-1} / y_{k-1}^{\mathrm{T}} y_{k-1}$
$z = H_k^0 q$
For $i = k-m, k-m+1, \ldots, k-1$
  $\beta_i = \rho_i y_i^{\mathrm{T}} z$
  $z = z + s_i(\alpha_i - \beta_i)$
Stop with $H_k g_k = z$
```

Figure4: Uphill Computation

List of qualitative bankruptcy parameters

Following are the few parameters or attributes using which bankruptcy can be predicted

Attribute Information: (P=Positive, A-Average, N-negative, and B-Bankruptcy, NB-Non-Bankruptcy)

1. Industrial Risk: {P, A, N}
2. Management Risk: {P, A, N}
3. Financial Flexibility: {P, A, N}
4. Credibility: {P, A, N}
5. Competitiveness: {P, A, N}
6. Operating Risk: {P, A, N}
7. Class: {B, NB}

By applying those first 6 qualitative parameters to the trained data set we can able to predict the whether the corporate firms will Bankrupt or Non-Bankrupt.

V. Implementation

Initially the data set will be on cloud. We will store it on local machine and say it as Resilient Distributed Dataset (RDD), the transformations of data return pointer to new RDD as RDD2.

The data set in RDD is in CSV format which will be converted to LibSVM format.

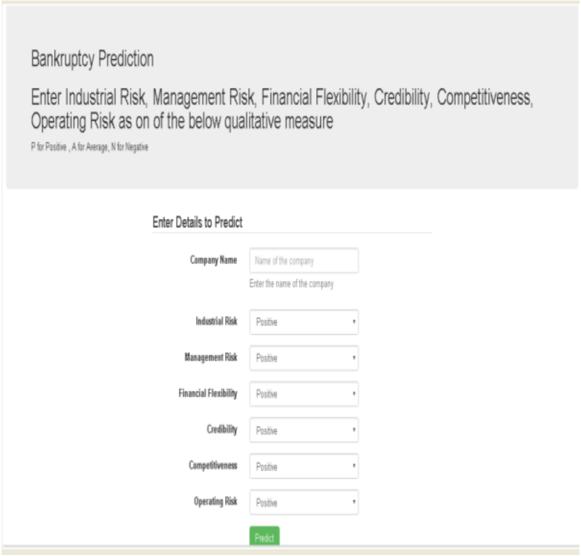
LibSVM allows for sparse training data. That is, the non-zero values are the only ones that are included in the dataset. Hence, the index specifies the column of the instance data (feature index). To convert from a conventional dataset just iterate over the data, and if the value of X (i,j) is non-zero, print j+1 : X (i,j).

The format of Lib SVM is as follows

<label> <index> **:**< value><index>**:**<value>

Then the dataset will be split into two partitions randomly in the ratio of 80:20 as training data set and test data set respectively. The model is build upon training date set and checked with test data set. Then the fresh data will be chosen by the auditor on the basis of given 6 features in the application to predict whether the corporate firms will bankrupt or non-bankrupt with already build model.

After predicting from the figure [4] we will get the precision as accuracy value depending on these six features.



Bankruptcy Prediction

Enter Industrial Risk, Management Risk, Financial Flexibility, Credibility, Competitiveness, Operating Risk as on of the below qualitative measure

P for Positive , A for Average, N for Negative

Enter Details to Predict

Company Name: Name of the company

Enter the name of the company

Industrial Risk: Positive

Management Risk: Positive

Financial Flexibility: Positive

Credibility: Positive

Competitiveness: Positive

Operating Risk: Positive

Fig 5 : Predictor

VI. Conclusion

The common ML tools used in BPM studies include ANN,SVM, RS, etc. Although these tools are unfit to directly analyse huge data, the Apache Mahout provides a suitable platform for some of these tools to use Big Data Analytics to solve classification problems. The problem of repeated intensive training on full data every time there is new data can be solved by using LBFGS. Overall, this work shows that Big data Analytics can be used for bankruptcy prediction by developing very robust BPMs. A framework is thus proposed for developing a Big Data Analytics based BPM.

VII. References:

[i] "The discovery of expertise decision rules from qualitative bankruptcy data using genetic algorithms"" by Myoung-Jong Kim, Ingoo Han.

[ii]"An Analysis on Qualitative Bankruptcy Prediction Rules using Ant-Miner" by A. Martin, T. Miranda Lakshmi, V. Prasanna Venkatesan

[iii] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, "Moa: Massive online analysis," The
Journal of Machine Learning Research, vol. 11, pp. 1601–1604, 2010.