TENTATIVE PAYMENT DATE PREDICTOR

SHIVAM, 18BCS3054

Department of Computer Science and Engineering, Student at Chandigarh University, Mohali, Punjab, PIN-140413, India

Email: badoniyashivam@gmail.com

# ABSTRACT

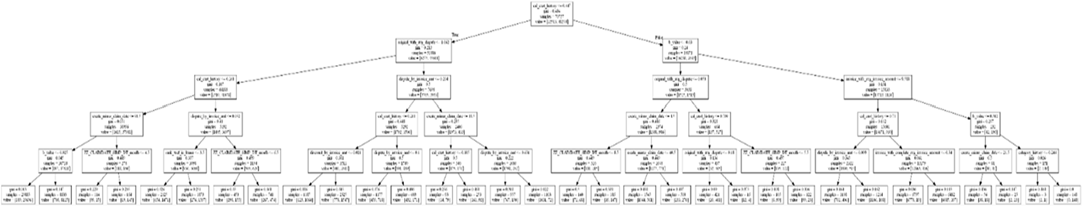
Keeping a steady cash flow is one of the biggest if not the biggest problem that Small to Medium Enterprises (SMEs) deal with daily. Within the different types of cash flow, Accounts Receivable (AR) classifies the balance of money that needs to be paid by the company’s customers. In the most typical case, after receiving goods or services, the customer receives an invoice with the amount that is owed to the supplier. However, this often does not happen before the aforementioned date, meaning that the invoice is often paid late. Intervention requires resources and over-intervention could cause unwanted customer dissatisfaction. Knowing whether an invoice is going to be paid late can be vital information. Current methods of late payment prediction focus only on the history between the seller and the buyer and are unusable when this history is not present. Intuitively, one’s business depends on the relationships and transactions that it has with its neighbors. Suggesting that neighbor behavior could be useful when predicting the cash flow of a company. Unfortunately, this type of information is not always given and needs to be data mining from non-relational data. This work presents a method for building a relational network of SMEs using entity resolution and improving the current state of the art of late payment prediction using features extracted from the graph.

# 

# 1. INTRODUCTION

The tentative payment date predictor is a B2B project which tends to predict payment dates from companies using previous data sets. This project uses machine learning models to construct logic on the data that is available from the previous records and construct a model which uses data for prediction. A B2B model is a type of business where a company doesn't deal with the real customers directly but it provides services to the companies and exchange business with them. This type of business run on a credit system which means that goods are given to the companies and shops on a postpaid basis and the amount of time a shop or company takes to pay the money in a given span of time affects their credit score in the market and their cap of product withdraw from the market. We have made this project for the companies giving products on credit that so that they can have safe play with companies that would tend to pay their loan on time and the companies who won't and this would help company in doing fair business with minimal loss.





**2. PROBLEM DEFINITION**

As explained before, our problem was defined as a binary classification problem to predict if either an invoice will be payed on time or late. Although we stated the problem as predicting classes, a wide range of models return probabilities instead of just labels. This is crucial in order to do a prioritization list and rank customers with higher chances of default. Also, since the model was planned to be deployed in a client that lately will need to retrain and update the model, it is important that we use a powerful model in terms of scalability, handle missing values, and would be easy do understand the results and retrain so the non-machine learning experts could have a sense about what is going on with the data

We tested our data with five different classification methods: Naive Bayes, Logistic Regression, k-Nearest Neighbors, Random Forest [4], Gradient Boosted Decision Trees [8]. Most of features came from historical data, for

example, sum amount late invoices, total invoices late and so on. In order to calculate these features for an invoice, we needed to define a period of time that we will consider to look back. This period is different from our trained dataset that defines which invoices we will consider. To define the best range of time to look back to calculate the features, we created a parameter that we call window size. In short, window size will be the number of months prior to an invoice that we will consider to calculate our features values. But, why not.

# 3. MODEL ARCHITECTURE

# Late payment prediction:

Late-payment prediction models are used on a more granular level, predicting when specific payments (i.e invoices) are going to default. While credit scoring is a well-researched topic, there has been very little research done in regards to predicting late payments

## **Financial transactions**:

Financial transactions can be modeled as a dynamic graph to analyze the interaction between different financial bodies. Work done by [33] explores different types of metrics in an economic system model as a complex network. The network explains monetary transactions between 105 clusters, each representing an economic activity standardized by the UN. The paper provides the following two contributions: A Network definition that is as follows:

Node, is an economic activity cluster, with the node weight being the summed transactions within the cluster.

An un-directed Edge is present when money flows between two sectors. Its weights show the summed money flow between two clusters in either direction.

## **Graph analysis and feature engineering:**

Graph embedding methods have seen a spike in interest and application in the last couple of years. These graph embeddings make it possible to encode graphs, making it possible to use graphs as input in various machine learning algorithms. Generally, the embedding algorithms are categorized by its method:

* Matrix Factorization: the embedding is achieved by factorization of the adjacency matrix.
* Random Walk based Deep Learning: uses the Skip Gram architecture to learn effective embeddings of random walks generated from the graphs.
* Non-Random walk Deep Learning: these methods leverage network architectures such as autoencoders or graph convolution layers to embed the input.

## **Feature engineering:**

Since networks cannot directly be used as input in machine learning models. The problem of prediction relies primarily on the quality of engineered features.

Therefore, it is important to have effective techniques that extract meaningful features from the networks. A well-known problem in this domain is the problem of link prediction, where the goal is to find missing links in a network using information about the nodes. Mutluetal.

## **Entity Resolution:**

Entity Resolution (ER for short, also known as Entity Matching, Entity Disambiguation, Record Linkage) describes the problem of finding unique entities from either single or multiple data sources. The paper done by Kondaetal. describes that while there has been an effort made in understanding the problem there is very little to no published work on ER in practice, end-to-end. The general outline of the paper is to show the methodology and workflow of doing ER in a real-world scenario. It argues that every unique case needs experts to differentiate the records and heuristics. The paper contributes a description of a real-world application, the goals set by the stakeholders involved and a description of the common ER challenges in real-world applications. The authors describe the first step to be setting up the matching rules:

1.Two records are a direct match if the unique ID is the same in both records.

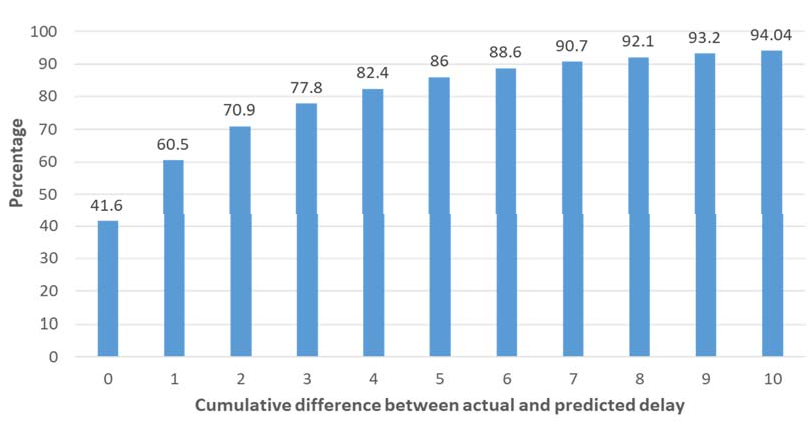
2.If Titles are similar.

3.If similar individuals are involved.

# INNOVATIONS IN PROJECT:

# 1.The given project can be turned into a full stack app or software where companies can input their data in CSV formats and using a simple UI, they will be able to predict the dates for the payment.

# 2.One more feature that we can innovate is the auto reminder feature: As one input their data in the software and predicted dates are months after so instead of checking everyday one can just process the data once and create a reminder sheet which will be set off to call or email companies for the payment reminder.





# The Problem with Averages

**What is Average Days Delinquent (ADD)?**

Average days delinquent (ADD) is the average number of days those invoices are past due - the amount of time between the invoice due date and the date it is paid. This calculation helps a company evaluate, along with other factors, the overall performance of collections department and their ability to convert accounts receivable to cash.

HOW TO CALCULATE AVERAGE DAYS DELINQUENT

1.Calculate average Days Sales Outstanding (DSO)

DSO = (Average AR / Total Credit Sales) x Number of Days

2.Calculate Best Possible DSO

Best Possible DSO = (Current AR / Total Credit Sales) x Number of Days

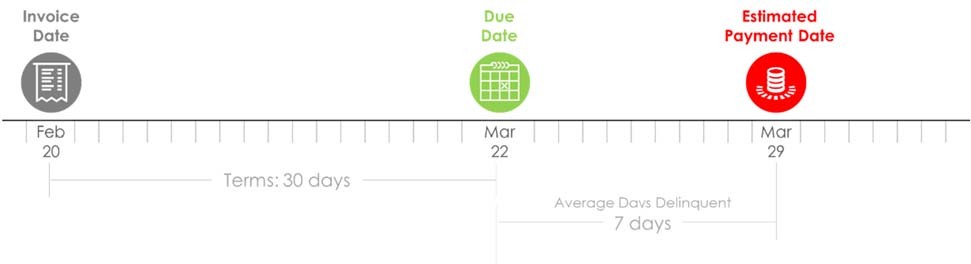
3.Calculate Average Days Delinquent

ADD = Days Sales Outstanding – Best Possible Days Sales Outstanding

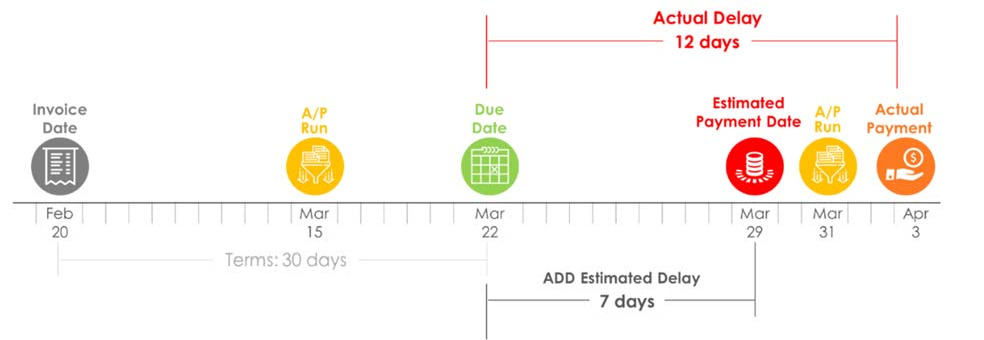
# Prediction with ADD: Fundamentally Flawed

In an attempt to unlock the strategic benefits of a proactive collections process, collections teams regard ADD as the current best metric to estimate payment date for a customer and consequently implement dynamic strategies and rules for proactive correspondence.

The subsequent figure illustrates payment date prediction using ADD as a metric for a small-to- medium business (SMB) customer where the A/P team runs its cycle in the middle or end of the month and the payment terms are 30 days. This is just one potential pattern with one customer where the A/P cycle plays a vital factor in payment.

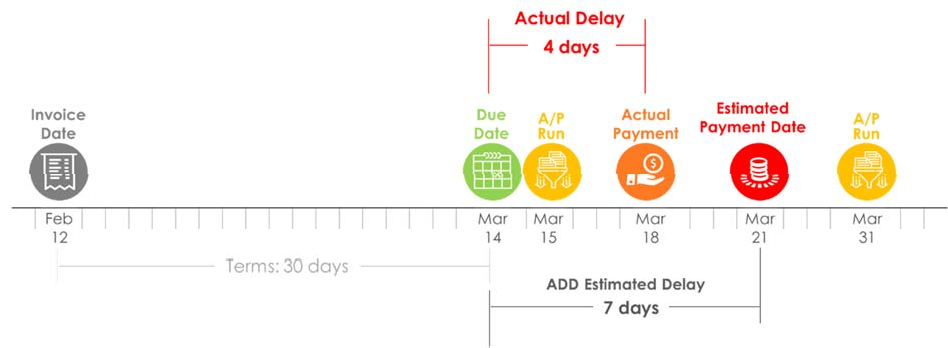


In the above scenario, the invoicing date is February 20th, and the due date is March 22nd according to the 30-days payment term. The ADD value of 7 indicates that on an average, the customer delays the payment by 7 days after the due date. Based on this estimation, the predicted payment date is March 29th. However, the A/P cycle schedule has not been considered in this assessment.



The dynamics of the real-world overshadow this prediction. The A/P cycle of the customer runs on March 15th, i.e., mid-month when the payment terms are still valid and the payment is not due, and therefore, the A/P team skips initiating the A/P process for this invoice. The next A/P cycle runs on March 31st when the invoice is already past-due. This is when the A/P team begins the invoice approval and payment process, and the actual payment comes in on April 3rd. As a result, the actual delay is 12 days as compared to the 7 days prediction delay.

In another scenario, the invoicing date is February 12th, and the due date is March 14th, while the predicted payment date, based on ADD of 7 days, is March 21st



**Embedding ML in Collections**

In this case, the A/P cycle runs on March 15th, **The ML Toolbox**

one day after the due date, and approves the Machine Learning being the hottest trend

payment for the invoice. As a result, the actual today, is incredibly powerful for predictions or

payment is made on March 18th with a delay calculated suggestions based on large amounts

of 4 days, as compared to the predicted of data. As businesses from many

payment date of March 21st with a delay of 7 days. industries have begun to rely more heavily on

machines to do the heavy lifting of A/R

The above scenarios clearly demonstrate that ADD processes, executives are able to

as a metric is not sufficient to predict payment deliver more value to their customers by

date and needs other customer-specific factors, reflecting on their roles and identifying

such as A/P cycle schedule, to predict the payment opportunities Machine Learning offer them.

accurately. Moreover, the above represents an

instance of a single customer. With the gigantic This section highlights the key algorithms

number of customers with which the collectors deal, explored for payment date prediction in the

a nearly infinite number of factors come into play. collections process. The following models

It is not practically feasible for collections teams to were considered and evaluated for the same.

identify each influencing factor and corresponding  **Linear Regression:** Linear regression

pattern for each customer to predict payment date is a linear model, i.e., a model

and tweak collections strategies. that assumes a relationship between the input variables(x) and the single

output variable(y). More specifically, that

# Why Machine Learning? (y) could be calculated from a linear

Combination of the input variables (x).

Evolved from the study of pattern recognition and  **Logistic Regression:** Logistic regression

computational learning theory in Artificial is a statistical method for analyzing a

Intelligence, Machine Learning explores the study dataset in which there are one or more

and construction of algorithms that learn from and independent variable which determine

make predictions on large volumes of data. It is the an outcome.

science of getting automations to act without being **Decision Tree**: Decision Trees

explicitly programmed to do so. are algorithms where the data is

continuously split according to a certain

Machine Learning could be leveraged to enhance parameter. The tree is explained by two

collections process as it enables payment date entities called decision nodes and leaves.

predictions that are up to four times as accurate, The leaves are the decisions or the final

using historical A/R data. As discussed in the outcomes, and the decision nodes are

previous section, machine learning identifies where the data is split.

relevant variables and analyzes valuable patterns **Support Vector Machine**: A support

in the collections cycle to make an educated vector machine performs classification

guess on the payment date for each by finding hyperplane that maximizes the

customer - science that is practically impossible margin between two classes. The vectors

for humans. Machine Learning has the ability to (cases) that define the hyperplane are the

process, analyze, and identify patterns amidst the support vectors.

enormous volume of historical data available for Naive Bayes: In machine learning, naive

each customer. It is able to predict the payment Bayes classifiers are a family of simple

date at an invoice level for all customers and probabilistic classifiers based on applying

help the collections teams become proactive Bayes’ theorem with strong(naive)

through improved dunning strategies. independence assumptions between the

features.

In probability theory and statistics, Bayes’ theorem describes the probability of an event, based on prior knowledge of conditions that might be related to the event.

**k-nearest neighbors (KNN):** K nearest neighbors

is a simple algorithm that stores all available

cases and classifies new cases based on a similarity

measure (e.g., distance functions).

**Random Forest:** Random forests or

random decision forests operate

by constructing a multitude of decision trees

at training time and outputting the class that

is the mode of the classes (classification)

or mean prediction (regression) of

the individual trees.

**Gradient Boost & Adaboost:** These

are boosting algorithms.

Ada(Adaptive)boost(Boosting) is an

iterative process that fits a sequence

of weak learners on different weighted

training data. It starts by predicting an

original data set and gives equal weight to

each observation. If the prediction is

incorrect using the first learner, then it

gives higher weight to observations which

have been predicted incorrectly.

Gradient boosting minimizes the loss

of the whole system using the

Gradient Descent method.

**K-Means:** K-Means clustering is a type of

unsupervised learning, which is used

when you have unlabeled data

(i.e., data without defined categories

or groups). The goal of this algorithm is

to find groups in the data, with the number

of groups represented by the variable K.

**Dimensionality Reduction:** In

Machine Learning, dimensionality

reduction or dimension reduction is

the process of reducing the number of

random variables under consideration

by obtaining a set of principal variables.

The following describes some models in detail and explores how they integrate with

the collections process.

**1.Binary Classification Model**

Binary Classification is the task of

classifying the elements of a given set into

two groups (predicting which group each

one belongs to) on the basis of a

classification rules.

In terms of collections, this model predicts

delay by answering Yes or No to the

question: “will payment for a given invoice

will be delayed?”

# 2.Multiclass Classification Model

# In Machine Learning, multiclass

# or multinomial classification is the task

# of classifying elements of a given set

# into one of three or more classes on the

# basis of a classification rule.

# 

For payment date prediction, this

this model classifies invoices into three

or more buckets based on the number of

days an invoice is delayed.

**3.Random Forest Regression Model**

In machine learning, a regression

model is used to predict a

continuous-values output.

This model predicts an actual

payment date based on the features

such as delay ratio

(Number of Delayed Invoices/Total

Number of Invoices) and their

Importance.

# Conclusion

# In this paper, we built a model that computes the probability score of an invoice being overdue in the context of AR practices. This is critical when dealing with a very large set of invoices, which in turn requires collectors to rank customers and focus on those more likely to be delinquent. Our results are significant, with an accuracy of up to 77%. The model developed in our work will be able to help our client attain a better sense of its AR operations and take better actions, thus improving its cash flow. Our set of historical features is small and captures the customer behavior payment using temporal information to make better prediction. We demonstrated by our experiments that using the window size with a small number of months (3) we were able to deal with concept drift in the dataset. We also created a new prioritization list that is able to rank customers in a more realistic way, helping the client to optimize their resources with respect to daily action of the collectors.

# Finally, as real world environments are always in continuous flux, our features distribution are shifting as well. We noted that the current AR process has been modified over the last year, and, as it evolves, our model should accompany that. We thus recommend a continuous evaluation of the analytics results so as to keep track of accuracy metrics as well as AUC, as discussed in the results section. From time to time, it seems necessary to retrain the model since all processes suffer from what is called concept drift, i.e., the relationship between the features and the labels evolve over time and classical machine learning approaches consider only stationary data. Future work involves providing an additional microservice for building visual analytics components to support collectors in the task of identifying recent customers’ behaviors. This need came out during interviews with collectors and SMEs. In these interviews, they mentioned to have some knowledge about customers’ behaviors, for instance, that some clients always pay few days late or that a certain client is part of an industry domain facing financial challenges. Such visual analytics is planned to be delivered as part of the ranked list UI so that collectors can grasp the recent behavior of customers with respect to all paying activities of recent invoices. This should foster collectors contacting target clients by phone, for example. While a discussion and analysis of the interviews and the design of the visualization UI is out of the scope of this paper, the overall project sits in an activity that encompasses a broader sociotechnical arrangement, from the technical development (dealing with data processing, classification algorithms, and technology deployment) to the human practices (dealing with the understanding of the collectors activity, human perception when dealing with visualizations, and how to incorporate such technology as part of collectors daily activity). To conclude, in this project we managed to delivery a machine learning model that successfully predicts the probability of an invoice of being late in order to rank customers and subsequently prioritize collectors’ efforts by focusing on those more likely to be overdue. In doing so, this brings us closer to the data-driven paradigm of decision-making processes where companies rely on data to improve their activities and direct their business needs, thus saving resources and improving efficiency.

# References

# [1] Naoki Abe, Prem Melville, Cezar Pendus, Chandan K

# Reddy, David L Jensen, Vince P Thomas, James J Bennett,

# Gary F Anderson, Brent R Cooley, Melissa Kowalczyk, et al.

# Optimizing debt collections using constrained

# reinforcement learning. In Proceedings of the 16th ACM

# SIGKDD international conference on Knowledge discovery

# and data mining, pages 75–84. ACM, 2010.

# [2] Bart Baesens, Tony Van Gestel, Maria Stepanova, Dirk

# Van den Poel, and Jan Vanthienen. Neural network survival

# analysis for personal loan data. Journal of the Operational

# Research Society, 56(9):1089–1098, 2005.

# [3] Butler B. Smith T. Swift T. Williamson

# J. Scherer W. T. Bailey, D. R.

# Providian financial corporation: Collections strategy. In

# Systems Engineering Capstone Conference. University of

# Virginia, 1999.

# [4] Leo Breiman. Random forests. Mach.

# Learn., 45(1):5– 32, October 2001.

# [5] Ricardo Cao, Juan M

# Vilar, and Andr´es Devia. Modelling consumer credit risk

# via survival analysis. SORT: statistics and operations

# research transactions, 33(1):0003–30, 2009.

# [6] Michelle

# LF Cheong and Wen SHI. Customer level predictive

# modeling for accounts receivable to reduce intervention

# actions. 2018.

# [7] Lore Dirick, Gerda Claeskens, and Bart

# Baesens. Time to default in credit scoring using survival

# analysis: a benchmark study. Journal of the Operational

# Research Society, 68(6):652–665, Jun 2017.

# [8] Jerome H.

# Friedman. Stochastic gradient boosting. Computational

# Statistics & Data Analysis, 38(4):367 – 378, 2002.

# Nonlinear Methods and Data Mining.

# [9] Gareth James,

# Daniela Witten, Trevor Hastie, and Robert Tibshirani. An

# Introduction to Statistical Learning: With Applications in

# R. Springer Publishing Company, Incorporated, 2014.

# [10] Maurice George Kendall. Rank correlation methods. 1948.

# [11] Elisa T. Lee and John Wenyu Wang. Statistical

# Methods for Survival Data Analysis. Wiley Publishing, 4th

# edition, 2013.

# [12] Michal Rychnovsk`y et al. Survival analysis as a tool

# for better probability of default prediction. Acta

# Oeconomica Pragensia, 2018(1):34–46, 2018.

# [13] Janika Smirnov et al. Modelling late invoice payment

# times using survival analysis and random forests techniques. PhD thesis, 2016.

# [14] Tarun Tater, Sampath Dechu, Senthil Mani, and Chandresh Maurya. Prediction of invoice payment status in account payable business process. In International Conference on Service-Oriented Computing, pages 165–180. Springer, 2018.

# [15] Bashar Younes. A Framework for Invoice Management in Construction. PhD thesis, University of Alberta, 2013.

# [16] Sai Zeng, Prem Melville, Christian A Lang, Ioana Boier-Martin, and Conrad Murphy. Using predictive analysis to improve invoice-to-cash collection. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 1043–1050. ACM, 20