VICTOR CHIMANKPAM NWAOGU CUSTOMER CHURN PREDICTION FOR BUSINESS INTELLIGENCE USING MACHINE LEARNING NEU 2019

CUSTOMER CHURN PREDICTION FOR BUSINESS INTELLIGENCE USING MACHINE LEARNING

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

by
VICTOR CHIMANKPAM NWAOGU

In Partial Fulfilment of the Requirements for the Degree of Master of Science In Information Systems Engineering

NICOSIA, 2019

CUSTOMER CHURN PREDICTION FOR BUSINESS INTELLIGENCE USING MACHINE LEARNING

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

by

VICTOR CHIMANKPAM NWAOGU

In Partial Fulfilment of the Requirements for The Degree of Master of Science In Information Systems Engineering

NICOSIA, 2019

Victor CHIMANKPAM NWAOGU: CUSTOMER CHURN PREDICTION FOR BUSINESS INTELLIGENCE USING MACHINE LEARNING

Approval of Director of Graduate School of

Applied Sciences

Prof. Dr. Nadire Çavuş

We certify this thesis is satisfactory for the award of the degree of Masters of Science in Information Systems Engineering

Examining Committee in Charge:

Assist. Prof. Dr. Boran ŞEKEROĞLU	Committee chairman, Department of Information Systems Engineering, NEU
Assoc. Prof. Dr. Yöney Kırsal EVER	Committee member, Department of Software Engineering, NEU
Assoc. Prof. Dr. Kamil DİMİLİLER	Supervisor, Department of Electrical & Electronics Engineering, NEU

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name:
Signature:
Date

To my family…

ACKNOWLEDGMENTS

I would like to sincerely thank my supervisor Assoc. Prof. Dr. Kamil DİMİLİLER for his understanding, patience, and guidance throughout my thesis journey. His supervision was paramount in providing a well-rounded experience to complete this research. I graciously thank Assist. Prof. Dr. Boran ŞEKEROĞLU my advisor, for stirring me towards the right direction during my graduate program.

Furthermore, I would like to thank my family for their love, prayers and their great confidence in me, also without forgetting my awesome friends for their genuine support.

ABSTRACT

The telecommunication industry is a one of the rapidly growing industry in recent times due to technological advancement and new life style adaptation to ubiquitous internet as well as communications. However, due to the ferocious competition among the telecommunication companies, churn prediction and management is, by far, one of the highly ranked challenges these companies encounter. There are however a variety of machine learning techniques utilized to predict likely customer who will churn from a telecommunication company to another. This thesis sort to solve a classification problem, in which customers who are likely to churn and those who will not where supposed to be predicted from the Tel-data data set. To achieve this, SVM (linear, RBF, polynomial and sigmoid kernels), MLP (with Adam, SGD and LBFGS algorithms) and Neural Networks (with Adam optimization technique) were employed and their results compared to choose which technique best fit the problem. Results showed that neural network with Adam optimization technique outperformed the other techniques listed.

Keywords: Churn; customer retention management; Machine learning; Multilayer perceptron; neural networks; Support Vector Machines.

ÖZET

Telekomünikasyon endüstrisi, teknolojik gelişmeler ve her yerde bulunan internete ve iletişime yeni yaşam tarzı adaptasyonu nedeniyle son zamanlarda hızla büyüyen endüstrilerden biridir. Bununla birlikte, telekomünikasyon şirketleri arasındaki şiddetli rekabet nedeniyle, kayıp tahmini ve yönetimi, bu şirketlerin karşılaştığı üst düzey zorluklardan biridir. Bir telekomünikasyon şirketinden diğerine geçiş yapacak olası müşteriyi tahmin etmek için kullanılan çeşitli makine öğrenimi teknikleri vardır. Bu tez, Teldata veri kümesinden çıkma olasılığı olan ve olmayacak müşterilerin tahmin edilmesinin beklendiği bir sınıflandırma problemini çözmek için sıralar. Bunu başarmak için, SVM (doğrusal, RBF, polinom ve sigmoid çekirdekler), MLP (Adam, SGD ve LBFGS algoritmaları ile) ve Sinir Ağları (Adam optimizasyon tekniği ile) kullanıldı ve hangi tekniğin soruna en uygun olduğunu seçmek için sonuçlar karşılaştırıldı. Sonuçlar, Adam optimizasyon tekniğine sahip sinir ağının listelenen diğer tekniklerden daha iyi performans gösterdiğini gösterdi.

Anahtar Kelime: Çalkalama; Müşteri Tutma Yönetimi; Makine öğrenme; Çok Katmanlı Algılayıcı; Nöral Ağlar; Vektör Makineleri Desteklemek.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

SVM:	Support Vector Machines
MLP:	Multilayer Perceptron
NN:	Neural Network
SGD:	Stochastic Gradient Descent
ML:	Machine Learning
BI:	Business Intelligence
CNN:	Convolutional Neural Network
RBF:	Radial Basis Function
LBFGS:	Limited Memory Broyden-Fletcher-Goldfarb-Shanon

CHAPTER 1

INTRODUCTION

The telecommunication industry is one of the rapidly growing industries in recent times due to technological advancements, new life style adaptations to ubiquitous internet as well as communications. Due to the ferocious competition among telecommunication companies, churn prediction and management is, by far, one of the highly ranked challenges these companies encounter (Hung et al., 2006). There are however a variety of machine learning techniques used to help retain customers to a mobile company.

Vishnukumar et al. (2017) in their study described Machine Learning, ML, which stems from Artificial Intelligence, AI, as a tool for data mining, and this is only a high-level intuition. Furthermore, when you think of various applications of data mining (Speech Recognition, Image Processing, Fraud Detection, and more) based on their objectives, which are, in most cases, peculiar with specific machine learning algorithms, you would come to understand that Machine Learning plays a vital role during the iterative cycle of data mining (also data science). The digital era has birthed numerous technologies, including Data Mining and Machine Learning (Artificial Intelligence) that are in use today. The telecom industry is one characterized by "Big Data", from customer records, which makes it feasible for mobile operators to carry our various research to predict customer churn, employing such machine learning algorithms as: statistical machine learning algorithms (Linear and Logistic Regression, Random Forest, Decision Trees, K-Nearest Neighbor), neural networks algorithms (Multi-Layer Perceptron), these lead to ascertain patterns from data. These patterns help in classifying data, making predictions and, eventually, building a suitable model from that specific data. (Huettmann et al., 2018).

Customer churn describes the loss of a customer to competitor(s) of the same service you provide (Mobile Telecom Service in this case), and it amounts to: loss of profit for Telecom Operators, loss of referrals from continuing customers, and more. Customer churn has become a worrying issue for service delivery business ventures (who place immense value on their customers). One of the most predominant sources of revenue (profit) of

telecommunication companies are their customer assets (Noe et al., 2017). Customer life cycle in had me noticed to revolve around the following phase: acquisition, build up, peak, decline and churn (Hudaib et al., 2015).

Countries who have adopted a liberalized telecoms sector (that is, lowering entry barriers for Greenfield Operators) experience intense competition such that it costs about 5 times more to win a new subscriber to retain an existing one, according to marketing literature; on the other hand, in developed countries which have adopted the "Porting Authorization code" (PAC), a customer could churn and retain their phone number thus raising the odds for a customer to churn. For this reason, Telecom Operators are on the market searching for optimum customer retention schemes. Customer churn can be grouped into involuntary churn (customers who are disconnected by the telecom operator for such reasons as death of a subscriber, fraud, bad debt or under-utilization) and voluntary churn (customers who initiate the churn themselves; it could be: deliberate due to pricing, poor customer service, network problems or incidental due to financial contingencies, location or major life changes) (Jahanzeb and Jabeen, 2007).

Hans Luhn, an IBM engineer, in 1995, birthed the concept of Business Intelligence (BI) and he described business intelligence as "the ability to apprehend the interrelationships of presented facts in such a way as to guide actions towards a desire goal" (Naumann and Herschel 2010). The last three decades have witnessed privatization of free markets, including the telecoms markets, in many countries leading to fierce competition in the telecom market. Business Intelligence comes as handy tool for those telecom companies who are hungry to be major players in the industry and has become, if not, the major task for any decision-making process in such firms. In this era of Big Data, business intelligence could be computationally expensive, but, on the other hand, it is the best choice for Customer Retention Management. In order for telecommunication operators to gain advantage in their competitive industry, they have adopted "personalized marketing" as an alternative to "conventional marketing". This is to assist them satisfy customers, predict churn and to avoid it so as to raise profit. This ultimately ameliorates investor confidence (Ekaterina, 2016). This process (decision making process in customer churn management) could be broken down into two: predicting customers who are about to churn from database; invent strategies to act on in order to gain competitive edge and avoid customer turnover. Intuitively, a solution for the first step is a pre-requisite for the next, this is where business intelligence

takes the lead role (that is, provide solution to the first process) so that effective strategies (could be promotional offers, mouth-watery deals) are implemented to minimize customer churn.

Neural networks, a subset of Machine learning, have proven to produce better and accurate results by employing optimization methods in comparison with statistical algorithms (owing to their non-linear mapping ability, robustness, wide range of optimization algorithms to choose from and precise prediction).

1.1 Problem to be Solved

Customer life cycle in had me noticed to revolve around the following phase: acquisition, build up, peak, decline and churn. One of the major challenges telecommuting companies face is their inability to study the patterns and behaviors of their customers from the big data they generate from their customers. This, therefore, makes it difficult for them to predict customer churn resulting in loss of income for these companies.

1.2 The Aim of the Thesis

This research aims to build a predictive model, with a high level of accuracy (90 percent and above), to assist Customer Retention Managers of Telecom Operators unravel the former process, that is, predict which customer would likely churn from their network, in order to come up with strategies to retain such customers.

1.3 The Importance of the Thesis

This research will prove the superiority of deep neural networks over conventional machine learning algorithms (support vector machines in this case), with their respective "performance" (measure of accuracy) to predict customer churn for mobile telecom operators for prompt and necessary actions to be taken by these companies.

1.4 Limitations of the Study

It is not easy to approach telecom companies for data to carry out effective and efficient research due to bureaucracy. The researcher believes the research would have been better if data was accessed more easily and in larger amounts. Machine learning algorithms work better with high performance computers, this research was limited by the access to such computers.

1.5 Thesis Outline

The thesis comprises of five chapters. The chapters are arranged thus:

Chapter 2: Reviews current literatures related to Customer Churn, Business Intelligence, and Customer Retention Management; as well some related works. Some technical concepts are as well deliberated in this chapter.

Chapter 3: Chapter three describes the current research methodology; theories and applications of machine learning techniques employed in this thesis are explained.

Chapter 4: Presents and explains the experimental results obtained from this research and finally.

Chapter 5: Conclusions and recommendations for prospective future works are described in this chapter.

CHAPTER 2

RELATED WORKS AND CONCEPTS

2.1 Business Intelligence

Vercellis (2009) in his book defines business as undertaking a productive venture to satisfy someone's needs by means income is earned. In his book he explained that, data is accrued business activities with customers can be analyzed and mined with the help of methods to give rise to recognizable patterns and intelligence which can be used to enhance the business as well as intensify customer satisfaction. The writer differentiated data mining and business intelligence. He defined data mining as statistical and machine learning techniques that helps in creating models by which decisions are made whereas business intelligence are mechanisms and ways by which data is gathered, examined and kept in visual forms to assist decision making.

The survival of every business depends on continuous monitoring of its environment. This enables an organization evaluate her performance, swiftly adjust to the environment as well as her future goals. These include steps taken by the business to monitoring the industry and every stake holder involved. Data generated from customer activates ought to be sought in formats that would be easily and rapidly comprehended by the executives of the firm.

"The term Business Intelligence was first used in 1989 by Howard Dressner, then a research fellow at Gartner group, as an umbrella term to describe concepts and methods to improve business decision making by using fact-based support", business intelligence can be defined as a data-driven Decision support system, DSS, that combines the collection and storage of data, and knowledge management with analysis to provide input to a decision process (Negash & Gray, 2008). The introduction of automatic processes at the different operational levels in a business organization (namely: transactional, managerial and executive) with help from information systems gave rise to large volumes of data (at the different operational levels in a firm), business intelligence systems are built to disseminate actionable information/ knowledge for a given business level (in a firm), after series of

analysis/evaluation have been carried out on available data, to invent effective strategies in order to gain a competitive edge amongst competitors in any industry.

A research conducted by Larson and Chang (2016) summarized views of other researchers on the definitions of business intelligence.

Business intellegence can be described to provide a business an edge by gaining the requsit information in order for her to carry out day to day actives more effectively.

The researcher also outlined business inteliigence as tools, technologies, applications, used in a business to decipher decision making informaiton which acrued data can not provide. This, in his view, enhances decision making in vaious aspects to the business.

Alspaugh et al. (2018) in their research explained business intelligence explicitly by saying it is a process in which data collection and storage together with knowledge management generates reasons upon which business decision are made. In their research, they made mention of fact that business intelligence improves decision making processes, skills and technology.

Richards et al., (2019) expanded on the definition of business intelligence to be a subset of business that involves the use of application and tools together with infrastructures and practices to enable information access and evaluation to improve efficiency and decision-making. Business and IT collaboration, resulting in data, are the challenges in business intelligence delivery.

Creswell (2005) points out that a methodology is used in a system with a number of procedures, techniques and rules operate that system. Successful business intelligence methodology would concentrate on the knowledge value chain rather than software development as conventional IT development relies on. Research has shown that business intelligence is unsuccessful in waterfall life cycles and software development practices. The use of information gained provides organizations with a value of business intelligence, not software and hardware (Larson, 2009).

Popular stumbling blocks historically faced in business intelligence projects include: fuzzy requirements; misunderstanding of how information is produced and utilized; reliability of data not calculated or understood, source process constraints determine development and support levels; data interpretation-based developments; findings not

proven promptly; and lack of confidence between IT (information technology) and business intelligence projects. Although these problems exist, the need for information was driven earlier by the big data trend. Big Data is used globally to define huge, complex data sets that cannot be handled using conventional IT techniques and applications (Boobier, 2018).

In essence, business is made up of four systems; Transaction Support System, Management Information System, Management Information System and Decision Support Systems and Executive Support System

Transaction Support System: Data is generated by the day to day transactions and activities of the business. The transaction support system is a system in which these data is collected and stored.

Management Information System: the stored data is then passed to the management information systems where it is classified, sorted and analyzed to identify recognizable patterns to extract relevant information (knowledge is been managed at this stage)

Management Information System and Decision Support Systems: extracted knowledge is then passed to the management information system where information that can be acted upon is derived. Strategies and decision are then formulated and made respectively.

Executive Support System: the final stage of business intelligence where actions are taken.

The first three stages of business intelligence are achieved by employing data mining tools, strategies and techniques. Figure 2.1 is pictorial drawing illustrating the various stages of business intelligence.

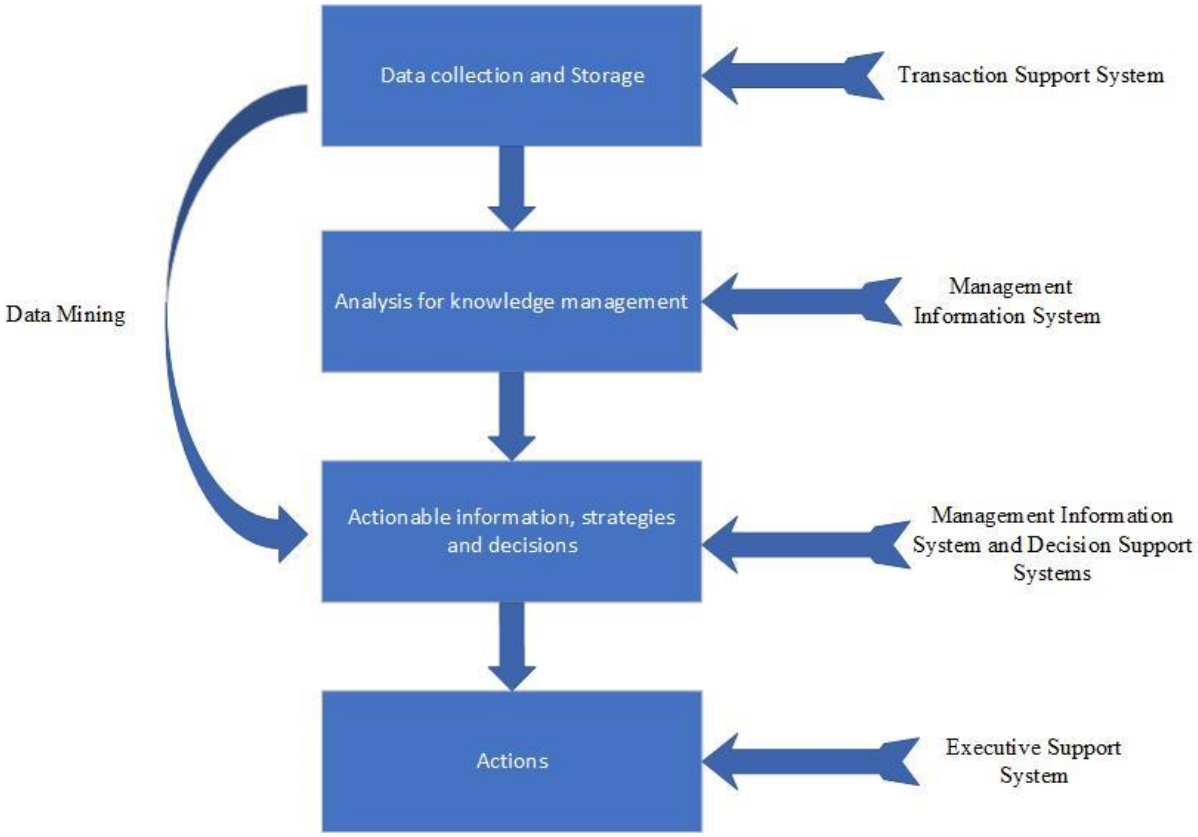

Figure 2.1: Components of business intelligence system

2.2 Data Mining

Data mining is a green-field and trending discipline whose applications cut across a good number of industrial sectors (that are characterized by Big Data). One of the motives behind Data Mining is the fact that Big Data contains vital and hidden information (in the form of patterns, classes or labels, clusters, structures, and many more) to be unraveled by owners of such data. We can, therefore, point out the basic function of data mining as the iterative process of discovering latent interesting patterns, and, or knowledge from Big Data. David Hand (2013) shed more light on data mining as an iterative process thus: “it is an ongoing process; one examines a dataset, identifies features of possible interest, and discusses them with an expert, goes back to the data in the light of these discussion, and so on”.

The process of discovering a useful data structure from a data set is defined as data mining. The framework could be multifaceted, including a set of rules, a graph or a network, a tree, one or more algorithms, and many more. A generalized data model is the knowledge gained

from a data mining session. The ultimate objective is applying new situations to models that have been discovered. Data mining as defined by SAS in Institute 1998 is a process of selecting, exploring and modelling large amounts of data to find and establish clear unknown patterns (Dittert et al., 2017)

Adhikari et al. (2016) described data mining as a set of methods to detect patterns and relationships in unknown data. Data mining can be categorized into two main methods; descriptive and predictive. The descriptive has to do with deriving meaningful patterns for decision making and outlining the properties of the data. Predictive, on the other hand, is the concept of using multiple variables in a database to forecast the value of other desirable unknown or future values in other to predict future behavior. Data mining techniques are in general used in communication, prediction, analysis and division of data.

In recent times, when businesses encounter problems, the problems are well defined in other to decide what mechanisms would be employed to solve these problems. Data relating to the problem is then collected and analyzed to map up the best fit model (test set). When this model is determined, it is tested and validated with some data (validation set) relating to the defined problem. Subsequently, a couple of strategies are drafted from these models for necessary actions to be taken. The entire process of defining the problem until a strategy is reached to solve the problem is known as Data mining. However, the tools, process and methodologies, used to collect data, analyze it to map up a model, test these models with the data and validate the results are collective known as machine learning. Figure 2.2 outlines the processes involved in data mining.

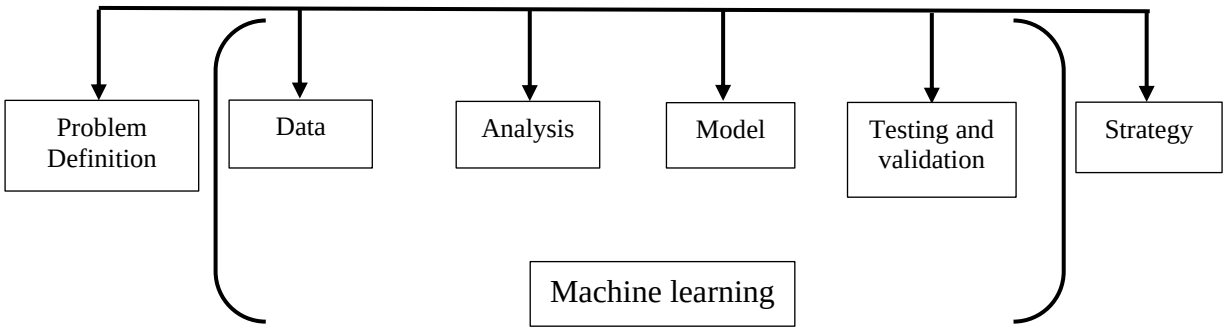

Figure 2.2: Data mining processes

2.3 Customer Retention Management / Customer Churn

For every subscription-based business model, churn rate is a health indicator for such businesses. The ability for a telecommunication company to retain her customers is currently one of the major challenges these companies face in recent times owing to the influx of competitors in the sector, which in essence grants customers the choice to move from a company to the other. That not with standing there is a notable relationship between customers' devotion, customers' loyalty, trust and churning in the telecommunication industry. A Telecom company's paramount goals have been to design product and services to suit the demand of their subscriber at affordable prices.

Simões (2016) outlined customer churn as a terminology used in the telecommunication industry to describe a customer's migration from one service provider to an alternative one. It could also be described as customers' lack of patronage of the service during certain times. The researcher also explained churn management as the service provider's means to preserve remunerative customers. Similarly, Ren et al., (2019) described churn management as a term used in the telecommunication industry to delineate steps taken to guard against losing the highly esteemed customers. The researcher summarized customer management as ability to predict the possibility of a profitable customer's decision to migrate form one telecommunication company to another, mapping up strategies to curtail this lost, implementing strategies to retain and eradicate the movements.

In essence, a company may divide their customers into two main groups; profitable and non-profitable costumers and focus turnover on acquisition from the profitable costumers, or categorize them by their "propensity to churn" and prioritize the effort to maintain productivity and churn inclination. The researchers mentioned that, a number of measures for profitability (e.g. current versus life-time, business compared to corporate, account versus customer, loyalty versus economic profitability, etc.) are still being standardized in the telecommunications sector (Manral, and Harrigan, 2018).

2.4 Related works

In the past, other researchers have employed couple of algorithms to obtain high accuracy for customer churn prediction in the telecom industry. Some have also employed hybrid algorithms (combining two or more algorithms) to predict customer churn. A related research is hereby conducted to outline a few algorithms used to predict customer churn, in telecom industry, by other researchers.

2.4.1 Customer churn prediction

Ismail et al., (2015) in their investigation realized that there has been a limited number of researches conducted on customer churn utilizing machine learning techniques, for which reason they decided to use artificial neural network to investigate and improve customer churn in the telecommunication industry. The research suggests the neural network method of Multilayer Perceptron to forecast customer churn in one of the biggest telecommunications companies in Malaysia. They methodologically compared results of Multiple Regression Analysis and Logistic Regression Analysis to predict costumer churn. They found a churn prediction accuracy of 91.28% made by neural network and concluded its superiority over the statistical models.

Mishra and Reddy (2017) in their study utilized Deep learning by Convolutional Neural Network (CNN) to carry out churn prediction in the telecommunication industry. They obtained experimental results indicating churn prediction accuracy of 86.85%, error rate of 13.15%, precision 91.08, recall 93.08%, F-score 92.06%. Their proposed methodology revealed that, the test results showed that the CNN is the best classifier in terms of all performance measures such as precision, errors rates, precision, recall and an F-score for the Churn prediction problem. They reemphasized that, the early churning forecast will avoid a loss for the company by forecasting consumer behavior. They as well proposed an improvement in the performance of churn prediction using deep learning by Convolutional Neural Network (CNN) by tensor flow framework with respect to time and accuracy.

Vafeiadis et al (2015) have also carried out a comparative study on the most common machine learning methods used in the telecom industry for the complicated problem of customer churn forecasting. All of our models were applied and assessed utilizing cross-

validation on a popular, public domain data set during the first phase of thier studies. Whereas the second phase examined the improvement in performance offered by boosting. They also ran a series of Carlo simulations for each techinque with a large variety of parameters in order to determine the most effective parameter blends. In their study, their results showed that the enhanced models are clearly superior to the plain versions (non-boosted). They additionaly discovered the best overall classifier, the SVM-POLY using AdaBoost with accuracy of almost 97% and F-measure of over 84%.

i. SVM

Rodan, et al. (2015) in their study outlined that, telecom firms have in recent times invested more in developing accurate forecasting models that could predict the customers who would end their subscriptions with them or switch to another competitor. The researchers used a Support Vector Machine (SVM) model to predict churn in the local telecommunications. In general, SVM models are parametric and the initial values of its parameters have great influence on its accuracy and performance. The researchers therefore combined evaluation metric while they tuned the parameters of SVM to maximize its effectiveness for churn management.

ii. Deep neural networks

Karanovic et al. (2018) in their study experimented with data provided by Orange Telecommunication Company to predict customer churning phenomenon. The researchers divided the study into various phases; removing null values (removal of missing values and redundant data), Lasso and manual feature engineering. They then applied CNN as classifier to classify preprocessed one-dimensional dataset with accuracy of 98.85%. In the researchers view, their proposed model could also be used to predict churning in areas of similar problem. Additionally, they mentioned that Ensemble techniques can enhance higher accuracy and AUC score. Lastly, they advised that problems are domain specific tool, and enhanced usability is often a deciding determinant of a system's success, therefore models must be altered to suit problems.

Whatever method is used, each problem is domain specific and better features are often the determining part of the performance of a system. Therefore, feature engineering is essential.

In the same vain Ahmed et al., (2019) conducted a research on customer churn prediction in the telecom industry utilizing Transfer Learning (TL) and Ensemble-based Meta-Classification. This, in their view, is a challenging task due to the large size of data, high dimensional features and imbalanced distribution of the data set. The proposed method "TLDeepE" was implemented in a couple of stages. The first stage used TL to refine several Deep Convolution Networks (CNNs), which are pre-trained. Telecom datasets are usually translated into 2D pictures by the matrix because deep CNNs have great image capability to understand. In the second phase, the projections of the Deep CNN are added to the initial feature vector and are therefore used to construct the final feature Vector for the GP and AdaBoost ensemble classifier. The performance of the proposed TL-DeepE system is compared to existing techniques using 10-fold cross validation for two standard data sets (Orange and Cell2cell.). The writers found out that, the prediction accuracy obtained was 75.4% and 68.2%, while the areas under the curve were 0.83 and 0.74 respectively.

CHAPTER 3

MACHINE LEARNING APPROACH

3.1 Machine Learning

Machine learning is born from the identification of patterns and the idea that machines can learn without being trained to perform specific tasks (that is, programs that learn without being directly programmed. As a result, learning is guided by data) with knowledge obtained through the ability to make successful choices based on the nature of the learning signal or input (Panesar, 2019).

Machine learning is an empirical and iterative (idea-model-test-idea) process that requires training a lot of models in order to obtain good performance on the goal of the problem, that is, find an effective model to fit the problem with minimal loss. Table 3.1 illustrates a tabular explanation for machine learning, its application and types of algorithms.

Figure 3.1 depicts machine learning as a subset of AI (artificial intelligence) and entails variety of algorithms to solve problems of different kinds. Thus, we can match Machine learning among the arms of AI (artificial intelligence), that supports modelling/building intelligence models, using computers, to make meaningful knowledge of data.

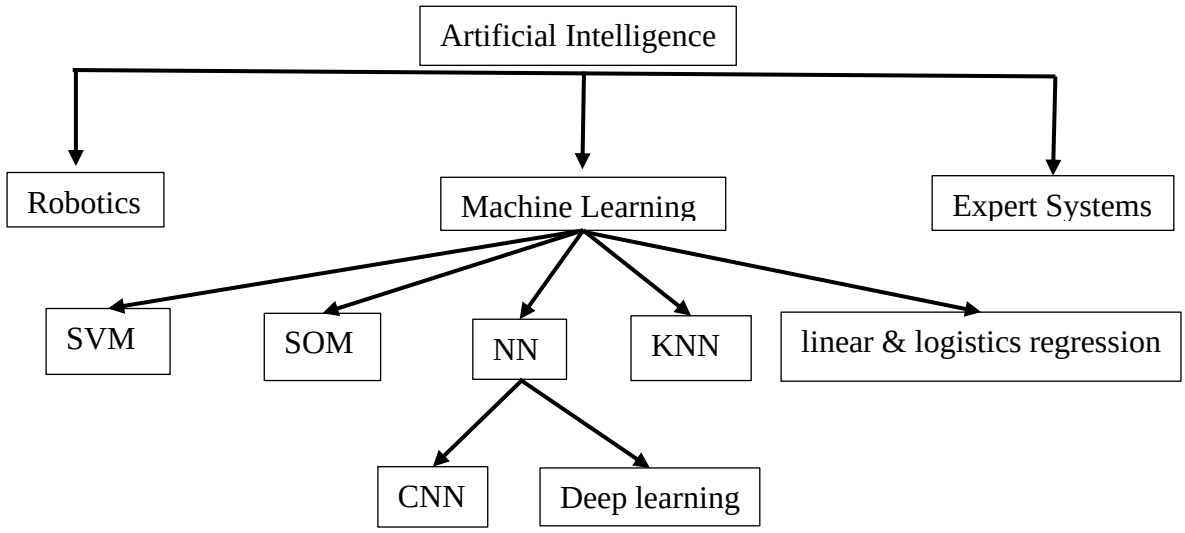

Figure 3.1: Machine learning as a subset of Artificial intelligence

3.1.1 Types of Machine Learning

Nowadays, machine learning can be utilized in every notable industry known and can also be employed to solve problems within unimaginable scopes but no matter how the problem is posed it can be grouped into the following types.

iii. Supervised learning

A supervised learning formula is such that, it has the ability to make observations (learn) from a labeled training data to help you forecast results from subsequent test data. You teach the computer using well labeled data in Supervised Learning (Chang et al., 2018). This is an indication that some, if not all, data are already marked with the right answer (label). Furthermore, supervised learning can be likened to where you have input and output vector variables, using an algorithm (pattern) we could learn the mapping function from input to output for the vectors. It's a matter of approximating the mapping function to estimate the output variables for that data if we have new input information. A data-science model takes time and technological know-how from a proficient data scientist group to successfully develop scale and execute reliable supervised machine learning. In reality, data scientists

need to reconstruct models to ensure that the results presented remain true until their technology improves.

In essence the process of algorithms learning from a trained dataset is known as supervised learning. Supervised learning enables a computer to grasp the mapping function from the input to the output with dependent and independent variables. The goal is to estimate the mapping function so as to forecast the output variables of these data when new input data are obtained.

iv. Unsupervised learning

In order to learn more about the data, unsupervised learning models the underlying and unknown structure in the data. Unsupervised learning is where the data are only input and no associated output parameters. (Label, class, value, etc.). Unsupervised learning sometimes described as self-organization, is a type of self-organized learning of Hebbian that helps to find newly discovered trends without existing labels in a data set. The principal component and cluster analysis are two of the most popular methods employed in unsupervised learning. Unsupervised learning can be used as a core method of analytics in the area of density estimation, while unsupervised learning includes many other dimensions that include data encapsulation and description. Supervised learning can be said to predict a conditional probability distribution depending on the input data label; unsupervised learning seeks to infer an a priori probability distribution on a given data. Table 3.1 below reports a summary of similarities and differences between supervised and unsupervised machine learning types, the first column has five characteristics that were used to describe their (supervised an unsupervised learning on the second and third columns respectively) similarities and differences.

Table 3.1: Supervised versus unsupervised learning

Characteristics	Supervised Learning	Unsupervised learning
Methods	Input variables and output variables are given	Only input data is given
Goals	To determine the accurate functions which will be used to predict output of when new dataset is given	To model the hidden patterns in a dataset in order to learn the details
Class	Machine learning problems, data mining problems and neural network	Machine learning, data mining problems and neural networks
Examples	Classification Regression Linear regression Support vector machine	Cluster Association k-means Association
Use	Utilized on image recognition, speech recognition forecasting, financial analysis and training neural networks and decision tree	Pre-processing of data during exploratory analysis as well as pre-training supervised learning algorithms

v. Reinforced learning

Reinforcement learning is a kind of "adaptive programming" in the field of artificial intelligence that trains algorithms using a reward and punishment system.

With continuous communicating with its surroundings, an agent learns to earn "rewards" for successful performances and "fines" for poor results. The agent learns for "optimizing" its reward and mitigating its punishment without human interference. A good analogy for reinforcement learning is when a child learns a new task (learning to walk for example), a huge contrast from supervised and unsupervised learning algorithms because, in this case, a reinforced learning algorithm is not exactly instructed on how to go about a task, but walks through the puzzle independently. As a model, (be it a self-driving car or a Chess game

program), it continuously interacts with its environment, a reward state is achieved based such performances as driving to a specified destination safely or a check-mate. Conversely, the agent or model draws a cost (penalty) for such performances as diverging from a specified destination or being checkmated.

Every supervised or unsupervised learning algorithm is modelled to solve a problem by either making predictions, unraveling patterns by means of classification, regression, or clustering approach. Table 3.2 below highlights all three machine learning algorithms (supervised, unsupervised and reinforcement learning) in the first column, the second column shows different machine learning problems where they could be employed and last column reports a few machine learning algorithms that could be utilized.

Table 3.2: Machine learning at a glance

Types of machine learning	**Machine learning problems**	**Notable machine learning algorithms**
Supervised learning	Classification, Regression	SVM, SOM, MLP, NN, CNN, Random Forests.
Unsupervised learning	Cluster	KNN, K-means Clustering
Reinforcement learning	Reward Function	NN

3.2 Classification

Classification, in machine learning, is the process of anticipating the faction (label, target, and category) of known data points. A model used for classification describes the approximation of function (f) that is able to map input features (X) to “discrete” output features (y) (Mahdavinejad et al.).

Spam detection, employed by providers of e-mail services, fits a classification problem. Here we have binary classification as there would only be 2 factions (spam, not spam). A classifier utilizes training data to understand the relationship between a set of input variables, and group them. So, a set of labeled emails would be exploited to train the data, and if the classifier is efficient enough, it could be utilized for detecting spam emails with new data.

Application of classifiers cuts across a wide range of industries, some of the notable applications of classifiers are in credit approval, medical diagnosis, target marketing, fraud detection.

i. Examples of classification

In general terms, making a prediction for a target that has discrete values is classification. For example, we may want to determine whether or not a blood sample's molecular composition (say its mass spectrum data) is cancer indicative; so, we have the problem of binary classification. Another instance would be a sample of mitochondrial DNA from which we would like to forecast the genus or species from where it originates; this type of classification is termed multi-class classification due to an extremely large number of categories from which to judge. And, if an instance may belong to more than one category then we are faced with a multi-label classification problem. The “goal” of classification problems, generally, is to equate an example correctly with one or more labels from a certain number of options.

By contrast, when trying to forecast a numerical value, the training goal is to find some equation to produce a reasonable guesstimate of the true (actual) value. Accuracy is judged, in this case, by how close the predicted value is to the expert's actual value, instead of whether the expected value is correct or not. This is regression, which usually requires techniques that are different from those used for classification.

Classification solves the problem of grouping variables based on their values (labels) while Regression solves the problem of predicting the outcome for a new entry.

3.3 Regression

Classification models determines what categorizes occurrence, regression models on the other hand predict a numerical value. Regression models are ML models utilized to estimate the relationships between variables. Regression specifically refers to the calculation of a continuous dependent variable or response from a set of input variables or functions within the ML and data science context.

We employ regression when the research goal is to find a formula that predicts a reasonable approximation of the true value if we try to forecast a numerical value. In this case, accuracy is rendered by how close the prediction is to the expert's actual value (true value), and not by whether the estimated value is identical or not.

In this research, our problem requires classification (we want to accurately predict potential churners of a telecom company using labeled data set from Kaggle comprises of data from customers who have either churned or not). But we would also carry out regression analysis on the problem.

There are a wide variety of methods for regression, from basic (linear regression) to advanced classical regression (Lasso or Elastic Net) up to more complex approaches such as gradient boosting and neural networks.

3.4 Cluster

In supervised learning, input data are observed and the objective features that need to be predicted from the training data while for clustering (an example of unsupervised learning) the dependent features are not provided in the case of clustering or unsupervised learning. Clustering aims to build a natural classification which can be used for data grouping, the concept underneath clustering is to separate data into clusters. Every cluster predicts the characteristic of future values of new data. Each group has an error predicting the forecast. “The prediction of the values for the features of an example is the weighted average of the predictions of the classes the example is in, weighted by the probability of the example being in the class.”

Cluster analysis is utilized to extrapolate algorithmic relationships to group data sets with similar attributes. It segments data that are not labeled. The cluster analyzes identify similarities in data rather than react to feedback and react based on the exact nature of those similarities in each new piece of data. This approach helps to detect abnormal data points that are not in any group.

3.5 Support Vector Machine

SVM has been explicitly defined as a distinguishing classifier utilizing a hyperplane for grouping data into classes. Vector support machines rely on the concept of decision planes characterizing decision thresholds. A decision-making plane distinguishes between objects belonging to different categories or classes. In essence, the algorithm when assigned labeled data, gives an optimal hyper-plane that classifies new examples in the case of supervised learning. A hyperplane could be a line in two dimensional spaces, which divides a plane into two parts with class fall on either sides of the line. However, in multidimensional spaces, SVM is required to determine a hyper plane to separate classes. Practically a multidimensional space describes a situation when the dependent variable ‘y’ has to be predicted with more than two independent variables ‘x’ (Longato et al., 2019). Noticeably, SVMs are being applied to solve problems in face detection, handwriting recognition, and many more.

3.5.1 Types of Kernel

i. Polynomial kernel

Polynomial kernel is a dynamic kernel and among the kernel functions associated with SVMs and “other kernel models”; it maps the similar vectors (referring to training samples) onto a higher dimensional plane over polynomials of the original variables, and enhances learning of non-linear models.

Intuitively, when trying to discover similarities of the input (training examples), the polynomial kernel considers the given features of the input samples as well as combinations of the input samples themselves.

Most often, such combinations are called interaction features in the field of regression analysis. Both polynomial Kernel and polynomial regression are characterized by the same feature space (without the combinatorial ambiguity on the number of parameters to be learned), so when input features are Booleans, they amount to logical association of these features. Polynomial kernels work better with normalized training data.

$$k(x, y) = (\alpha xT\ y + c)d \tag{3.1}$$

Modifiable parameters are: the slope "α"; the constant term "c" and the polynomial degree "d".

For degree-*d* polynomials, the polynomial kernel is construed as

$$k(x, y) = x^T\ y + c \tag{3.2}$$

where *x* and *y* are vectors in the input space, i.e. feature vectors obtained from training or test samples and $c \geq 0$ is a free parameter to strike a balance between higher-order against lower-order terms in the polynomial.

When $c = 0$ connotes a homogenous kernel. (Furthermore, a generalized polynomial kernel divides x^Ty by a user-specified scalar parameter *a*.

As a kernel, *K* describes the inner product in a feature space centered on some mapping φ:

$$k(x, y) = \big(\varphi(x), \varphi(y)\big) \tag{3.3}$$

The description of φ can be understood using an example. Let $d = 2$, so we get the special case of the quadratic kernel. After applying multinomial theorem (twice—the outermost application is the binomial theorem) and regrouping, we have

$$K(x,y) = (\sum_{i=1}^{n} x_i y_i + c)^2 = \sum_{i=1}^{n}(x_i^2)(y_i^2) + \sum_{i=2}^{n}\sum_{j=1}^{i-1}(\sqrt{2x_i x_j})(\sqrt{2 y_i y_j}) + \sum_{i=1}^{n}(\sqrt{2cx_i})(\sqrt{2cxy_i}) + c^2 \quad (3.4)$$

ii. Linear kernel

The linear kernel, said to be the most simplified kernel function, is expressed by the inner product <x,y> in addition to an optional constant **c**. Kernel algorithms, employing a linear kernel, are much alike to their non-kernel kind, i.e. KPCA with linear kernel is the same as standard PCA (Yuan, 2016).

Linear Kernels are used when the data has been observed to be linearly Separable (they can be separated using a single line), mostly employed for such applications as Text Classification (where there are a lot of features) as each alphabet is a new feature.

iii. Radial basis function

Radial basis function kernel, or RBF kernel, is a popular kernel function employed in support vector machine classification as well as various other learning algorithms that employ kernels for optimization. The RBF kernel on two samples x and x', represented as feature vectors in some input space, could be defined as:

$$k(x, x') = exp\left(-\frac{\|x - x'\|}{2\sigma^2}\right) \quad (3.5)$$

$\|x - x'\|$ stands for the squared Euclidean distance between the two feature vectors. σ is a free parameter. An equivalent definition involves a parameter $1/2\sigma^2$:

$$K(X, X') = \exp(\|x - x'\|^2) \tag{3.6}$$

The value of the RBF kernel decreases with distance ranging between zero (in the limit) and one (when x = x'), and has a ready to use measure of similarity and interpretation. The Kernel's feature space is characterized by an innumerable number of dimensions:

$$\begin{aligned}\exp\left(-\frac{1}{2}\|\mathbf{x}-\mathbf{x}'\|^2\right) &= \sum_{j=0}^{\infty}\frac{(\mathbf{x}^\top\mathbf{x}')^j}{j!}\exp\left(-\frac{1}{2}\|\mathbf{x}\|^2\right)\exp\left(-\frac{1}{2}\|\mathbf{x}'\|^2\right)\\ &= \sum_{j=0}^{\infty}\sum_{\sum n_i=j}\exp\left(-\frac{1}{2}\|\mathbf{x}\|^2\right)\frac{x_1^{n_1}\cdots x_k^{n_k}}{\sqrt{n_1!\cdots n_k!}}\exp\left(-\frac{1}{2}\|\mathbf{x}'\|^2\right)\frac{x'^{n_1}_1\cdots x'^{n_k}_k}{\sqrt{n_1!\cdots n_k!}}\end{aligned} \tag{3.7}$$

iv. Sigmoid kernel

Also known as the Hyperbolic Tangent Kernel and a kernel for Multilayer Perceptron (MLP), Sigmoid Kernel stems out of Neural Networks, where artificial neurons frequently adopt the bipolar sigmoid function as the choice mechanism for activation.

$$k(x, y) = \tanh(\alpha x^T y + c) \tag{3.8}$$

It is worthy of note that an SVM model implemented with a sigmoid kernel function can be likened to a neural network having two layers. This kernel gained popularity for SVMs owing to its origin from neural network theory and performance in practice, despite being only conditionally positive definite. When implementing this kernel, the slope, alpha, and the intercept constant C, are such parameters that can be tuned. For most applications, alpha is 1/N, (N represents data dimension). A more detailed study on sigmoid kernels can be found in the works by Hsuan-Tien and Chih-Jen.

3.6 Multilayer perceptron (MLP)

A multi-layer perceptron (MLP) is an artificial neural feedback network generating a bunch of outputs as a result of a set of inputs. MLP is defined by a number of layers of inputs connected to the output layers as a directed map. For training network, MLP uses back propagation. MLP is a neural network approach for learning in which signal goes through nodes in a unidirectional manner. All the nodes in MLP model has a non-linear activation function except the input nodes. MLP, like NNs, uses backpropagation to "train" the artificial neural network. It is worth mentioning, however, that MLP is a supervised learning technique.

Multi-layer perceptron is frequently employed in addressing computational neuroscience problems as well as parallel distributed computing problems, more generally supervised learning related problems.

MLP has gained relevance for such problems as machine translation, speech recognition, image recognition among others.

3.7 Neural Network Models

Neural Network (or Artificial Neural Network) is a complex adaptive (has the capacity to change its internal structure by adjusting weights of inputs) system that could learn from examples. ANN is an information processing model inspired by the biological neuron system (the human brain). A NN is characterized by a large number of highly interconnected processing elements, neurons, to solve problems and follow the non-linear path and process information in parallel throughout the nodes. Neural networks are designed to solve pattern recognition problems, which are straightforward for humans and demanding for machines, such as identifying and classifying pictures of cats and dogs, identifying pictures with numbers on them, with applications ranging from optical character recognition to object detection (Andreas et al., 2016).

The first mathematical model of neuron was developed by Warren McCulloch and Walter Pitts in 1943. In their study they defined a simple math model for a neuron, representing a single cell of the neural system that takes inputs, processes the inputs to return an output.

This model is called the neuronal System of McCulloch-Pitts (Lemley et al., 2017) and is defined by Equation 3.9 below.

$$Y = \sum (\text{weight} * \text{input}) + \text{bias} \tag{3.9}$$

An artificial neural network which was modeled from the human brain looks as in the figure 3.2 below. The input variables in Figure 3.3 are x1, x2.... xn; the inputs w1, w2.... wn are weights, b is the bias that is added to the weighted inputs in order to form net inputs; bias and weights are both modifiable parameters of the neuron. Parameters are fine-tuned by applying some learning rules (gradient descent). Neuron output can vary from -infinity to + infinity. The neurons do not have boundaries therefore we need a method to map between the neuron input and output. This projection process is called activation function.

There are many neural cells in the brain of humans for information processing. Each neural cell was seen as a simple processing unit that are connected to each other for electrical signal (information) transmission known as biological neural network. Neuron dendrites receive input signals from another neuron and respond on these inputs to an axon from another neuron.

Signals of other neurons are received from dendrites. The body of the cell summarizes all input signals in order to output a signal. Synapses are the point where neurons interact. Electric or chemical signals are disseminated to the next neuron.

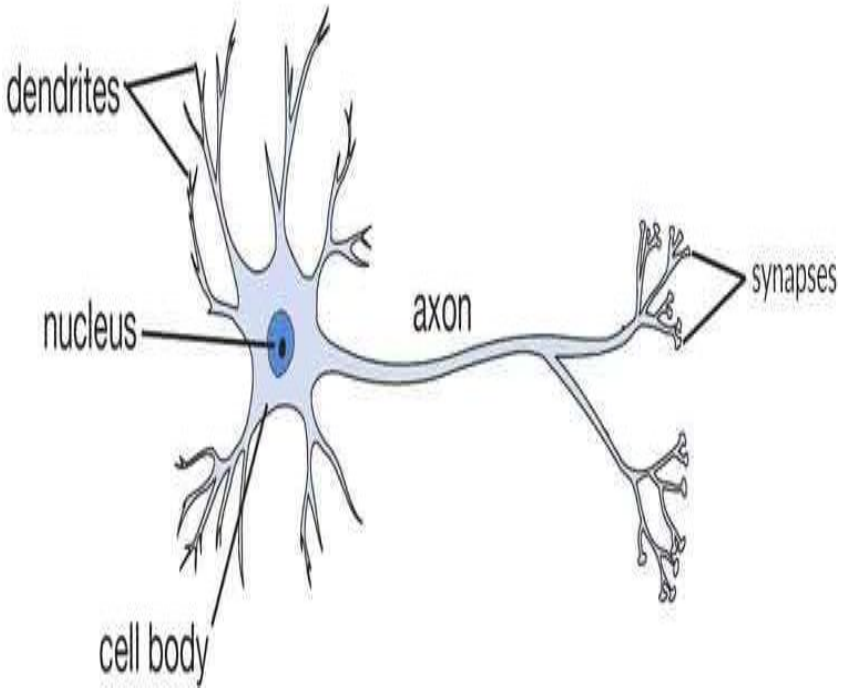

Figure 3.2: Biological Neuron (Zurada, 1992)

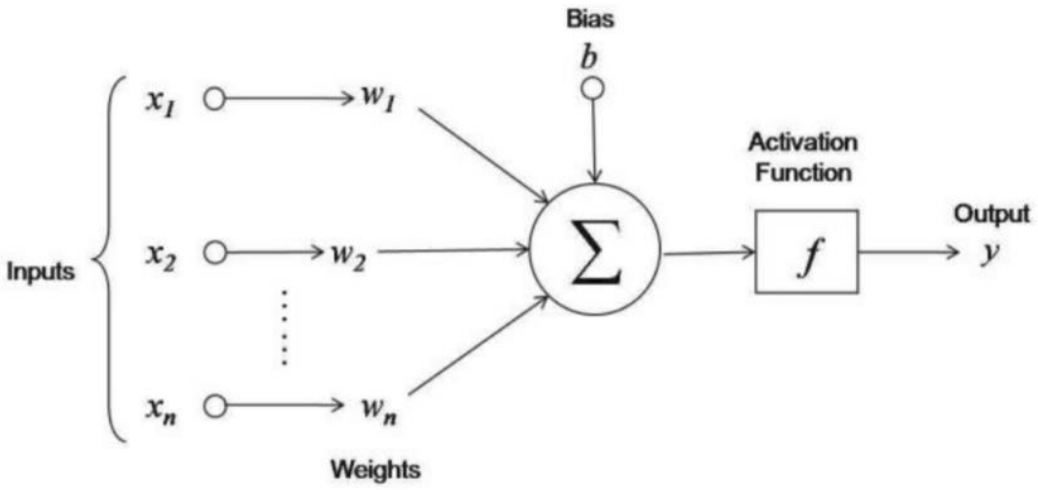

Figure 3.3: Mathematical neuron (Zurada, 1992)

3.8 Optimization Algorithms

Optimization is a fine-tuning process of any machine learning algorithm where a number of parameters are varied. This process is often tiresome but it aims at making the right selection of parameters to obtain a significantly high accuracy. A few numbers of optimization techniques have been exploited during this work.

3.8.1 Adam

Adam optimizer is considered as an ensemble of RMSprop (Root Mean Square propagation) and Gradient Descent with momentum optimization algorithms for NNs. In Adam, the squared gradients are employed to scale the learning rate (like RMSprop) substituting the gradient with the moving average of the gradient, like SGD with momentum.

i. Characteristics of Adam

In Adam, step size is practically restricted to the step size hyper-parameter in each iteration. This property gives an intrinsic insight into the previous hyper-parameter of inherent learning speed. Step size of Adam update regulation has no variation with the magnitude of the gradient, which assist when undertaking such areas as saddle points or ravines (SGD struggles to quickly navigate through them) with tiny gradients. Adam in intended to consolidate the pros of Gradient Descent with Momentum, which works better with sparse gradients, RMSprop works better when online setting is done properly. Having techniques allows us to utilize Adam for variety of tasks. Adam, as some would say, is the blend of RMSprop and SGD with momentum.

3.8.2 RMSprop

The focal point of RMSprop is to maintain the moving average of the squared gradients for each weight, and to divide the gradient using square root of the mean square. Hence its name, RMSprop (root mean square).

3.8.3 Gradient descent

Gradient descent is the algorithm which we employ to minimize the cost function (obtain a global minimum value- that is, the derivative of the cost function gets closer to zero). A global minimum (or local minimum, or local optimum) is the point where the gradient (derivative) of the cost function is equal to zero. Minimizing the cost function helps to achieve better accuracy (whether classification or regression problem) such that the predicted value (label) is same as the true label (in order words we're trying to increase the probability of getting the true label for a given example). This can be achieved by a simultaneous update of all parameters relating to the cost function.

i. Gradient descent with momentum

Instead of just relying on the current gradient in order to update weight (ω_{t+1}), the gradient descent replaces the current gradient with the Vt gradient, which stands for the velocity, represents the current gradient and the last gradients as their exponential moving average (Seppälä, 2019).

$$\omega_{t+1} = \omega_t - \alpha V_t \tag{3.10}$$

Where

$$V_t = \beta V_{t+1} + (1 + \beta)\frac{\partial L}{\partial \omega_t} \tag{3.11}$$

And *V* initialized to 0.

ii. Stochastic gradient descent

Stochastic gradient descent is a method of optimizing uncontrolled problems of optimization. Contrary to gradient descent, by considering a single learning example at a time, SGD approximates the true gradient. The first-order SGD training routine is enforced by the class SGD Classifier. The algorithm records the examples and updates the design parameters for each example according to the update rule provided by

$$\omega \leftarrow \omega - \eta(\alpha \partial R(\omega)/\partial \omega + (\partial L(\omega T xi + b, yi))/\partial x \quad (3.12)$$

Where:

η = learning rate which controls the step-size in the parameter space;

b = intercept.

Default learning rate for classification problems, (learning_rate = 'optimal') is given by

$$\eta^{(t)} = \frac{1}{\alpha(t_0 + t)} \quad (3.13)$$

t = time step

3.8.4 Limited-memory BFGS

Limited-memory (L-BFGS or LM-FGS) BFGS is a quasi-Newton optimization algorithm that requires only small amount of computer memory, it approximates Broyden – Fletcher – Goldfarb – Shanno (BFGS) algorithms. It is a popular algorithm for machine learning parameter estimation.

L-BFGS method is considered as an adaptation of the BFGS method to large problems, and the implementation of both methods is very similar. In the BFGS method, the approximation Hk to the inverse Hessian matrix off is updated by

$$H_{\mathrm{k+l}} = \mathrm{k} + \mathrm{V}_k^T H_k V_k + p_k S_k S_k^T \tag{3.14}$$

Where

$$V_k = 1 - p_k S_k S_k^T \tag{3.15}$$

$$s_k = x_{k+1} - x_k, y_k = \mathrm{g}_{k+1} - g_k, \text{and } \mathrm{P}_k = {}^{1}\!/_{y_k^T s_k} \tag{3.16}$$

The search direction is given by

$$\boldsymbol{p_{k+1} = -H_{k+1}\, g_{k+1}} \tag{3.17}$$

3.9 Regularization

Regularization is a process that adjusts the learning algorithm significantly so that the model is more generalized. This also increases the performance of the system with respect to the new dataset. This is a regression that limits, regulates or reduces the estimates of the coefficient close to zero. This technique, in other words, deprecates studying a more complex or versatile model to escape the risk of overfitting. The fitting process requires a loss function, known as extra sum of squares or RSS. The coefficients are picked, such that they minimize this loss function. The main merit of regularization is its ability overcome the issue of bias variance trade off, so we do not experience either under fitting or over fitting of the data set.

3.10 Confusion Matrix

A large number of classification models produce a probability number for the dataset, for binary classification problems the probability number is between 0 and 1, instead of a target variable like YES/NO (churn/non-churn for this research). The next logical step is to convert this probability number in the model to check the model's performance (accuracy) in comparison with the target / dependent variable, the predicted result could be compared to actual result and summarized as follows:

a) True Positives: Observations where the actual and predicted results were true

b) True Negatives: Observations where the actual and predicted results weren't false

c) False Positives: Observations where the actual results were false but predicted to be true

d) False Negatives: Observations where the actual results were true but were predicted to be false.

Confusion matrix is a suitable representation of this summary as shown in the figure 3.3 below.

True Positives (TP)	False Negatives (FN)
False Positives (FP)	True Negatives (TN)

Figure 3.4: Confusion matrix

CHAPTER 4

IMPLEMENTATION AND RESULTS

4.1 Implementation

This chapter focuses on the methods and techniques applied on this thesis as well as obtained results after the application of selected techniques on the dataset. This research seeks to prove the superiority of Deep Neural Networks over Conventional Machine Learning algorithms (Support Vector Machines in this case) for predicting customer churn with a dataset.

4.1.1 Techniques Used

Machine learning is an empirical and iterative (idea-model-test-idea) process which requires training a lot of models in order to obtain better accuracy, that is, attain an effective model to fit the problem.

4.1.2 Dataset

The Tel dataset, coped from “kaggle.com”, consists of 7043 instances (customers), 21 variables (attributes).

The dataset was normalized and rescaled in to a binary [0, 1]. The following are attributes that make up the dataset.

1. Customer ID was a combination of alphabets, numbers and strings that was unique to every customer, a primary key in other words. It was observed that there was mutual dependency between customer ID and the remaining variables (one to many relationship).
2. Gender consisted of two distinctive categories, a customer is either a male or a female. Either end of these categories (male, female) was assigned with a discrete value before they were furnished to any of the algorithms.

3. Senior citizen made up of two discrete (binary) values so this variable did not require to be scaled further, a senior citizen (valued = 1) is a customer over the age of 65 years while non- senior citizens (valued = 1) are customers under the age of 65 years.

4. Partner as an attribute was made up of two categories: married customers were labelled as “Yes” while single customers were labelled as “No”; that were further scaled to binary values.

5. Dependents was coined for the purpose of little tax exemptions here and there, the feature comprised of two specific categories that were labelled thus: dependent customers are such that have been enrolled as dependents “someone’s” tax form and labelled as “Yes” while the opposite are customers who have dependents themselves labelled as “No”; and will undergo further scaling.

6. Tenure was measured in months; this feature described the length of time a customer’s subscription has been active on the network. A continuous numeric data with saddle and peak points at one month and seventy-two months respectively.

7. Phone service was another feature that would put up with further scaling into discrete binary values, a categorical data (yes or no) that described customers who had active phone service or did not have.

8. Multiple lines categorized customers into: customers who either had multiple lines or did not, and customers who did not subscribe for phone service, when scaling into scaling binary value data wrangling was employed such that customers without an active phone service and customers who didn’t subscribe for multiple lines are treated the same.

9. Internet service was a distinctive feature with a “one to many relationship” with the following six variables (online security online backup, device protection, tech support, streaming TV, streaming movies) this feature specified what category of internet service a customer subscribed to, the choices are: Digital Subscriber Line

(DSL), Fiber Optic or none of the above. During data wrangling DSL and fiber optic categories were treated as one entity while the other option (NO) was treated as another entity.

10. Online security described customers who paid for protection from attacks while their devices stayed online and comprised of Yes, No and No internet categories, categories No and No internet are treated as one entity while the remaining category is treated as a different one during data wrangling.

11. Online backup outlined customers who either subscribed to have their selected data stored on the company's cloud storage platform (Yes) or not and those with inactive internet service (no internet service), the last two categories belong to one entity and the first category belongs to a separate entity during data wrangling.

12. Device protection represented those customers who signed up to have their devices replaced or fixed in the case of an accident, had three categories viz: customers who had device protection (Yes) or not and customers who did not subscribe for internet service (No internet service), the last two categories belong to one entity and the first category belongs to a separate entity during data wrangling as well.

13. Tech support distinguished customers into such categories as: those who didn't consent to free tech support from their provider (No) and those who did, as well as those customers who did not pay for internet service, the first category was handled as one entity while the second and third categories belong to a separate entity during data wrangling.

14. Streaming TV reported customers into three categories such as: customers who did not pay to watch live streams from TV channels (No) (treated as a separate entity during data wrangling), those who paid and, of course, those who did not subscribe for internet service. The last two categories were treated as one entity when data wrangling was performed.

15. Streaming movies reported customers and their corresponding categories in like manner as Streaming TV, and these categories were handled accordingly during data wrangling.

16. Contract showcased various payment plans available to customers, the payment plans fell into the following categories: monthly, annually and biannually. The foremost plan (monthly) was treated as a single entity while the remaining two were handled as one for further scaling.

17. Paperless billing portrayed two discrete mode of payment which customers opted for and did not require further scaling. Customers either chose paperless billing (YES) or did not.

18. Payment method was characterized by three categories named: Electronic payment method, mailed check or a bank transfer. Mailed check and Bank transfer categories were further scaled as one (customers who paid via this method chose did not agree with Paperless billing) while electronic payment, on the other hand, was scaled as a separate entity from the other two.

19. Monthly charges depicted how much money (dollars) was accrued by the network provider (monthly) from every customer on the dataset based on what services they subscribed for with the network provider, a continuous numerical data with values ranging between 18 and 150 dollars and had to go through further scaling.

20. Total charges reported the total value, in dollars, earned by the provider from each customer with respect to how their tenure (that is, every value represented monthly charged multiplied by the tenure a customer spent with the network). Again, this was a continuous numeric data that had to be scaled before it could be fed into any algorithm.

21. Churn was our target value (dependent variable) and was characterized by these two categories: customers who have churned (customers lost to competitors) labelled as

Yes and customers who did not. It was pretty easy to wrangle this attribute as each label got a binary number assigned to it (one for churners and zeros for non-churners).

4.1.3 Jupyter notebook

Jupyter Notebook web application allows editing of python codes and equations. It also gives a visual representation, modification of these codes. It is used in data science, for numerical simulation, ML, information transformation and statistical modeling. Jupyter Notebook supports python and creates comprehensive documentation after execution of the python codes.

4.1.4 Python

Coding in an appropriate environment would enable short lines of codes. Python is such a powerful programming language enriched with numerous library functions to making it easier to program complex problems with a fewer line of codes compared to other programming languages. A number of python ML related libraries were imported in other to successfully implement in other to successfully implement this research.

- Imblearn
- Pandas
- Keras
- Statistics
- Matplot lib
- Scikit_learn
- Num_Py
- TensorFlow
- Pandoc
- Seabourn

4.1.5 Structure of the Developed Model

The developed model was made up of various machine learning techniques as explained in Figure 4.1 below. It is composed of load libraries on python, import telecom data exploratory Data Analysis and outlined the various stages in the development of the models.

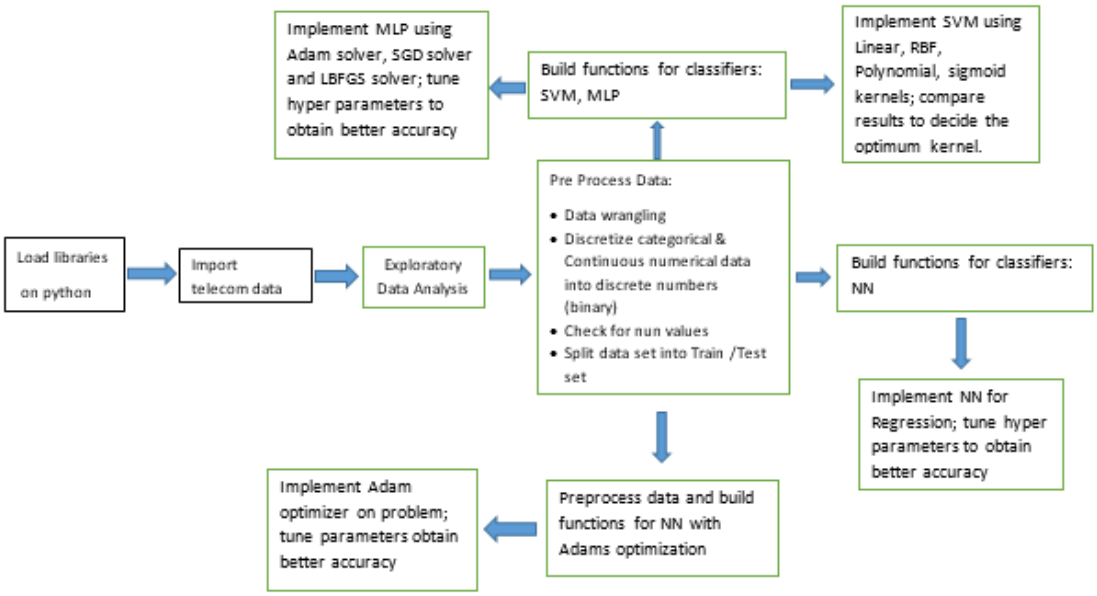

Figure 4.1: Block diagram of the model

i. Load libraries on python

To ensure successful implementation of the machine learning models, python libraries were imported as mention in figure 4.1 above.

ii. Import data

Tel data dataset from “kaggle.com” was imported in csv format on the Jupyter Notebook. This command was executed to enable data to be imported and ready for exploitation using python on the python notebook.

iii. Exploratory data analysis

It can be observed from the figure 4.2 below that we have an imbalanced class problem with our data set, there is a significant gap between the number of customers who churned from the company and customers who did not. A little over 5000 customers remained with the network company while the number of customers who churned from the network was a little below 2000.

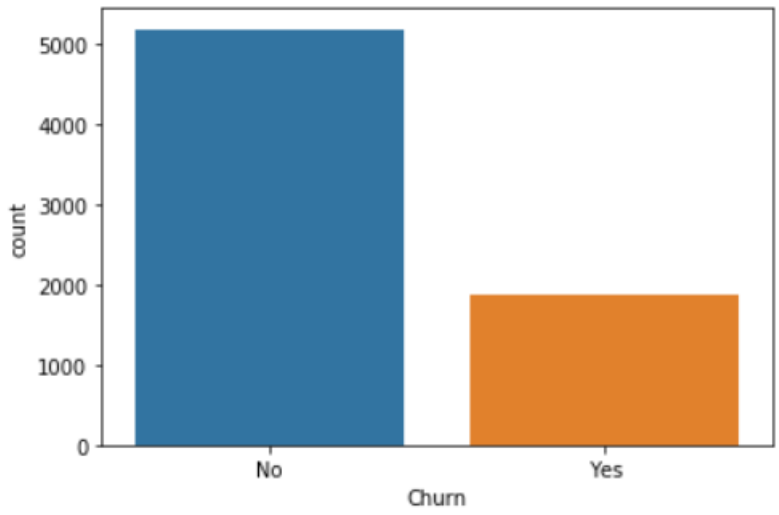

Figure 4.2: Count plot of number of churners

Figure 4.3 below is a bar chart comparing the number of customers who churned based on their gender (male or female). The chart shows that the number of customers who did not churn (which was over 2500 respectively for both genders) was significantly more than the number of customers who churned (about 800 customers for both genders). A customer's gender did not have much relationship with the dependent variable (Churn) because same number of customers from both gender categories churned from the network we can infer that gender has no relationship churn.

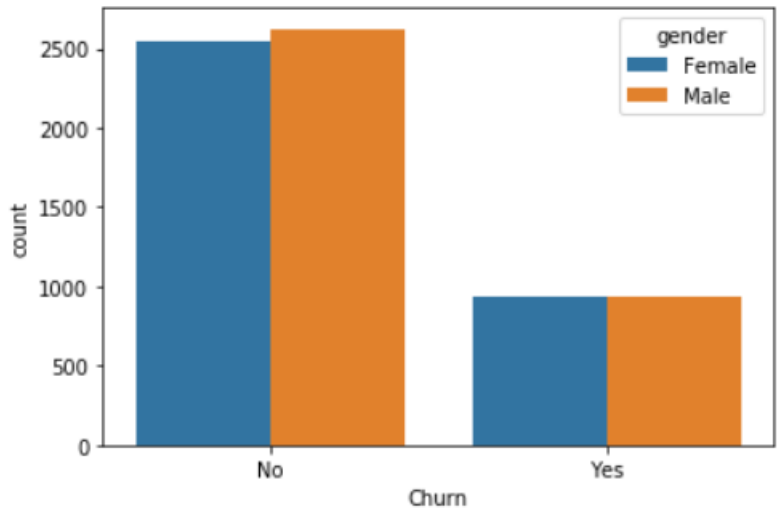

Figure 4.3: Count of number of churners based on gender

Customers who did not subscribe for Internet service did not also have Online Security, Phone protection, Online Backup and Tech Support, neither did they Stream on either of the streaming platforms (TV or Movies). Thus, we can infer that the variable, Internet Services, comes as a package. Also, there were customers who had no phone service (say for calls) and subscribed to either of the Internet Service Options (DSL, Fiber Optic), and, the customers who had no internet service subscription used the Network's phone service. These observations can be seen from the next ten bar charts shown below.

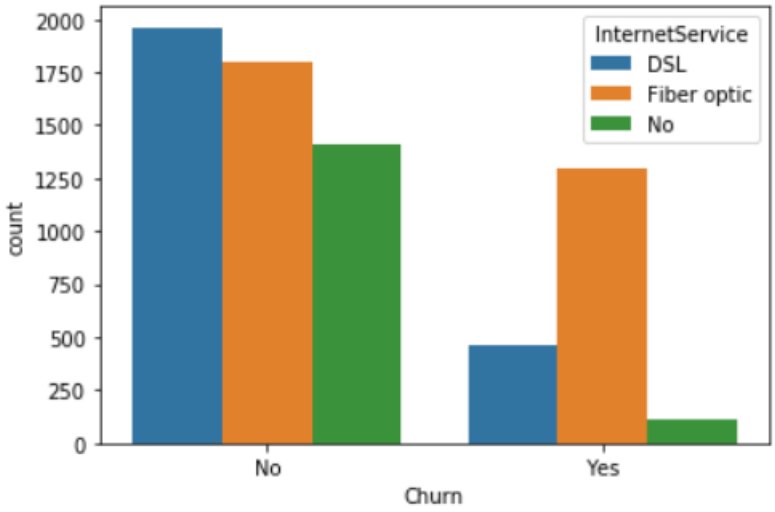

Figure 4.4: Count plot for customer who used phone service and internet

Figure 4.4 above is a bar chart showing the number of customers who subscribed to both phone and internet services. Over 3000 customers used both phone and internet service (fiber optic) and about 600 subscribers used digital subscriber line (DSL) type of internet service without phone service.

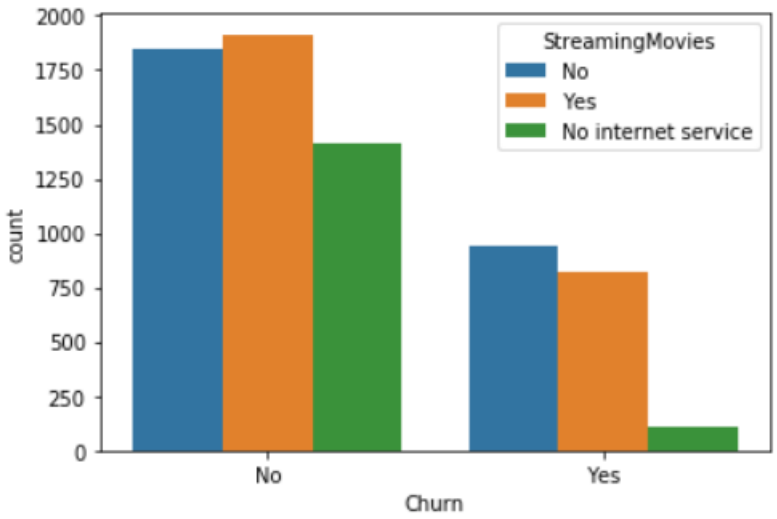

Figure 4.5: Count plot for number of churners who subscribed to streaming movies

Figure 4.5 above compares the number of customers who churned from those who did not based on if they had subscribed to streaming movies, or not, or did not have internet service. Over 1750 customers who either subscribed for streaming movies or not did not churn, while less than 250 customers without internet service churned from the network.

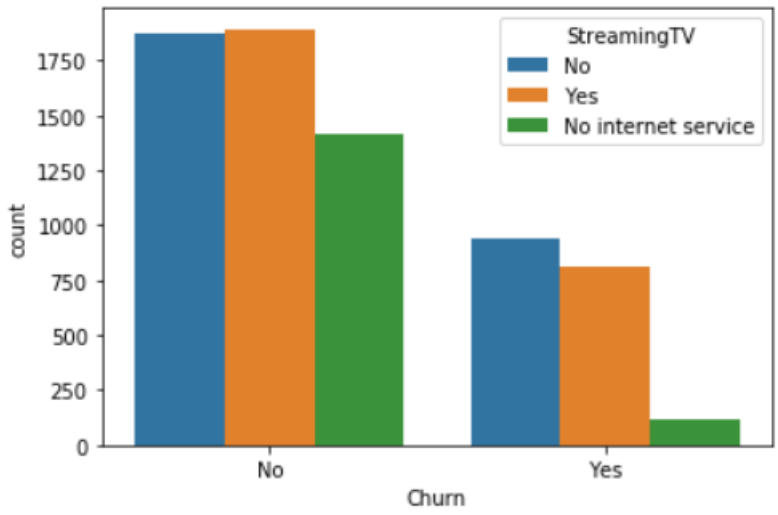

Figure 4.6: Count plot for number of churners who subscribed to streaming TV

Figure 4.6 above displays a bar chart comparing the number of customers who churned from those who did not based on whether they had subscribed to streaming TV, or not, or did not have internet service. Over 1750 customers who either subscribed for streaming TV or not did not churn, while approximately 100 customers who did not pay for internet service churned from the network.

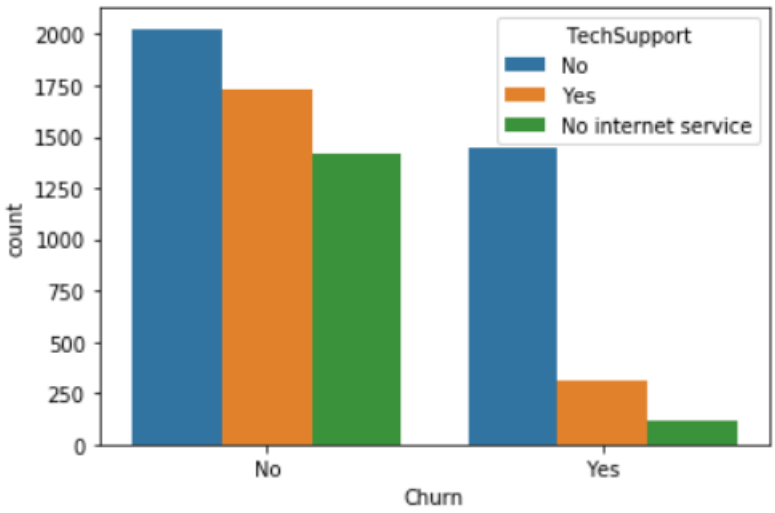

Figure 4.7: Count plot for number of churners who subscribed to tech support

Figure 4.7 above is a representation of a bar chart comparing the number of customers who churned from those who did not according to the following categories: customers who paid for tech support, those who did not and customers without internet service; on a bar chart. Overall, the chart clearly shows that customers who did not churn, collectively (that is regardless of their subscription status), is significantly greater than the number those who churned. We can also see from the chart that 2000 customers without tech support stayed with the network provider while more than 1250 of them left the network.

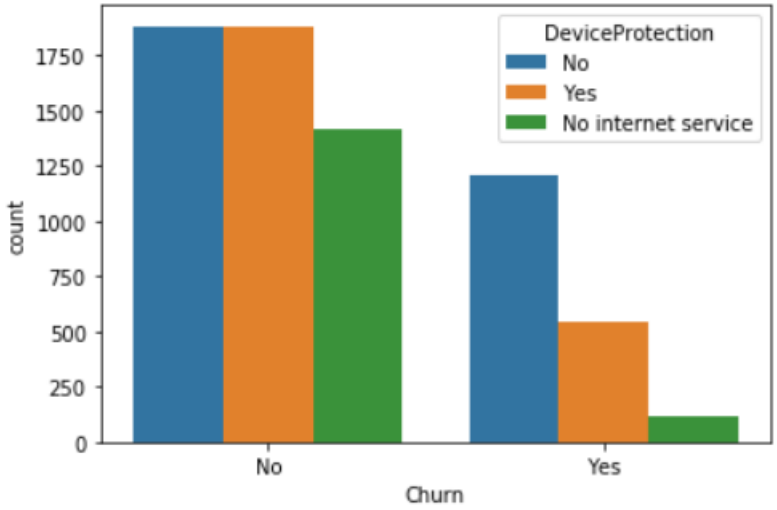

Figure 4.8: Count plot for number of churners who subscribed to device protection

Figure 4.8 above displays a bar chart comparing the number of customers who churned from those who did not based on whether they had subscribed for device protection, or not, or did not have internet service. While about 1900 customers who either subscribed for device protection or not did not churn, there was a significant difference between the number of customers who paid for device protection (roughly 500) and customers who did not pay for this feature (about 1100) who eventually churned.

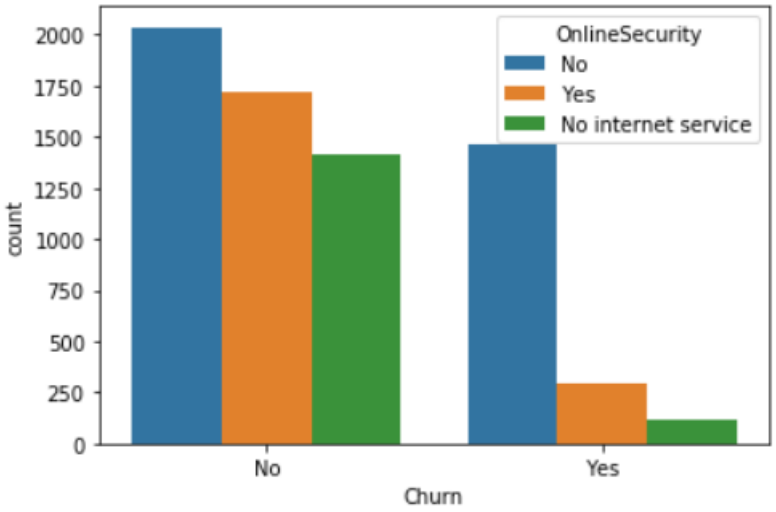

Figure 4.9: Count plot for number of churners who subscribed to online security

Figure 4.9 above is a representation of a bar chart comparing the number of customers who churned from those who did not according to the following categories: customers who paid for online security, those who did not and customers without internet service; on a bar chart. We can see from the chart that 2000 customers without online security stayed with the network provider while more than 1250 of them left the network.

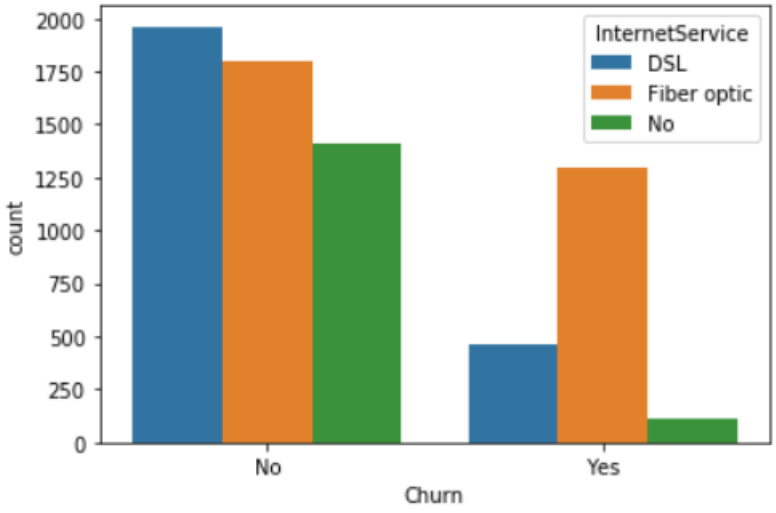

Figure 4.10: Count plot for number of churners who subscribed to internet service

Figure 4.10 above shows a bar chart comparing the number of customers who churned from those who did not based on what type of internet service they subscribed for (digital subscriber line, DSL, fiber optic, or no inter service). While about 1900 customers using DSL and over 1750 customers paying for fiber optic internet service did not churn, there was a significant difference between these numbers for customer who eventually churned; there were about 400 DSL subscribers and over 1250 fiber optic subscribers who churned from the network.

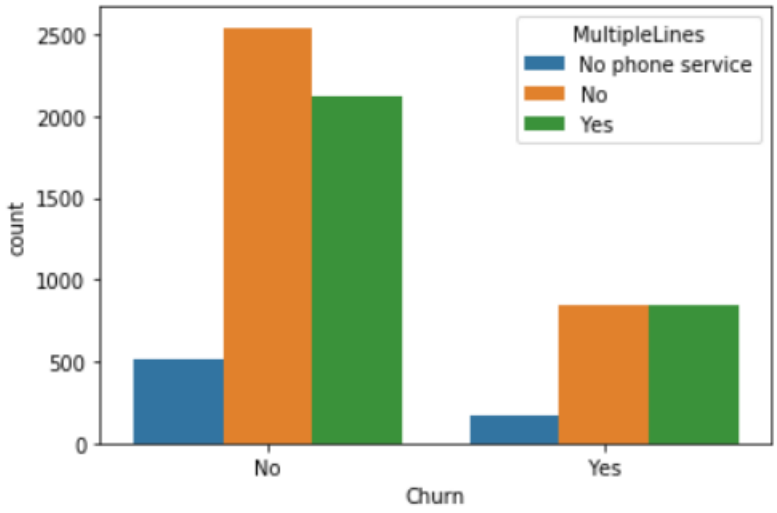

Figure 4.11: Count plot for number of churners who subscribed to multiple lines

Figure 4.11 above displays a bar chart comparing the number of customers who churned from those who did not based if they had multiple phone lines, or not, or did not subscribe for phone service. Over 2500 customers without multiple phone lines did not churn, while approximately 750 was the number of customers who did not pay for multiple lines and those who did respectively that churned from the network.

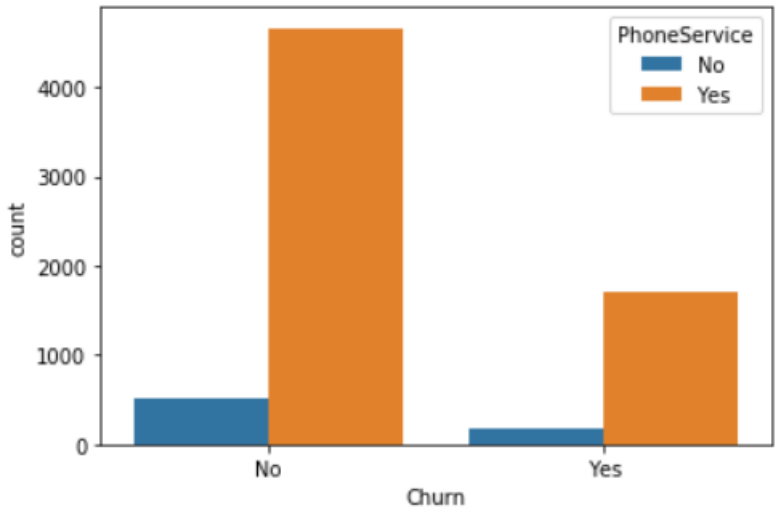

Figure 4.12: Count plot for number of churners who subscribed to phone service

Figure 4.12 above is a bar chart comparing the number of customers who churned and is dependent on whether they paid for phone service or not. The chart shows that the number of customers with active phone service who did not churn was over 4000 which is significantly more than the number of the same category of customers who churned, approximately 1500.

It is also clear that customers on a monthly payment plan were more likely to churn than others (i.e. customers on either annual or business intelligence annual payment plans). Most customers who churned made payments through Electronic Check.

There are no “null values” (missing entries for different variables so we could either remove such the column- if it contains a lot of missing values such that assigning values would not be a good approach- or assigning values to for the missing entry) in the Tel data set from “kaggle.com”, and the figure below shows our data set is “clean” (no missing entry in any column). A clean dataset would help our model perform effectively because it has adequate values. Figure 4.13 shows there are no null values in the dataset.

```
customerID          0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
tenure              0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV         0
StreamingMovies     0
Contract            0
PaperlessBilling    0
PaymentMethod       0
MonthlyCharges      0
TotalCharges        0
Churn               0
dtype: int64
```

Figure 4.13: Number of null values

iv. Data Pre-processing

Either of our models require feature (variables) extraction because they (the algorithms that were utilized) carry out this tasks themselves under the hood. Figure 4.16 represents the function used in processing the Tel data set.

SMOTE is an oversampling method. This operates, instead of making copies, by producing artificial samples from the minor group. The algorithm selects two or more similar instances (using a distance measure) and disturbs one instance at a time by a random sum within the variance of the adjacent instances.

v. Build function for classifiers: SVM, MLP

Telecom_churn_prediction function was defined to fit the training set on a model, make prediction on the model with the test set. Furthermore, calculations of the probability of accuracy on the predicted value in comparison with the true (actual) value.

vi. Functions for NN

A Deep neural network is a neural network- has the framework of a neural network- with "many" hidden layers. Below are the basic functions (building blocks) to model a NN.

Initialize Parameters: The weights and bias parameters, W and b respectively, were initialized using random initialization. Size of hidden layers (that is number of hidden layers) is four.

Define Activation Function: The activation function of choice, sigmoid in this case is defined as a sub-function

Define Forward Pass: Forward propagation is defined which would be called on at a later time.

Compute cost: Cost function for gradient descent to achieve a global minimum is defined in this section.

Back pass: This section defines the functions that would perform gradient descent (back propagation).

Update parameters: An optimum value for the learning rate is required in order reduce the computational time for gradient descent to converge to a global minimum.

Predict: This section defines the functions that would perform prediction using the test set.

vii. Adams optimization algorithm

Adam is an optimization algorithm that can be used to update network weights iteratively based on training data instead of the conventional stochastic gradient descent procedure.

Straight forward to implement. Computationally efficient. Little memory requirements.

4.2 Results

4.2.1 Implementing Multi-Layer Perceptron

MLP itself is a feedforward neural network as its name suggests. Stochastic gradient descent (SGD) is an iterative method for optimizing an objective function with suitable smoothness properties (e.g. differentiable or subdifferentiable). It can be called a stochastic

approximation of gradient descent optimization as it substitutes the real gradient (calculated from the whole data set) with an estimation of it (calculated from a randomly selected data subset). 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 prominent for parameter estimation in machine learning.

Batch gradient descent was used with all three solvers; Adam, SGD, LBFGS. It was observed that the optimal batch size for SGD algorithm was "auto" (the algorithm chooses an optimal batch size itself), SGD algorithm is likened to batch gradient descent.

Table 4.1: Results from MLP using SGD optimizer

Batch Size	**No. of Layers**	**No. of Hidden Layers**	**Max. No. of Iterations**	**Accuracy**
2500	1	100	500	75
Auto	3	300	500	78
Auto	5	500	500	80

Table 4.1 above shows results retrieved from implementing MLP using SGD optimizer which recorded 80 percent accuracy with 5 hidden layers (each layer has 200 neurons). This model performed poorly when several batch sizes were applied (with one hidden layer) but performed better by giving it the liberty of choosing an optimal batch size. Furthermore, increasing the number of hidden layers in the model gave rise to better results.

Table 4.2: Results from MLP using LBFGS optimizer

Batch Size	No. of Layers	No. of Hidden Layers	Max. No. of Iterations	Accuracy
3400	1	100	500	88
3400	2	200	500	89
400	3	300	2500	90
400	4	800	3000	91

Table 4.2 above shows results retrieved from implementing MLP using LBFGS optimizer which recorded 91 percent accuracy with 4 hidden layers (each layer has 200 neurons). This model showed great performance as several batch sizes were applied (with one hidden layer) but performed better by adding more layers to it.

Table 4.3: Results from MLP using Adam optimizer

Batch Size	No. of Layers	No. of Hidden Layers	Max. No. of Iterations	Accuracy
Auto	1	100	200	83
1800	2	200	500	88
750	4	800	2000	90
2000	5	1000	1000	91

Table 4.3 above shows results retrieved from implementing MLP using ADAM optimizer and it recorded a peak accurate value of 91 percent accuracy with 5 hidden layers (each layer has 200 neurons) and mini batch size of 2000. This model performed very well, like LBFGS when several batch sizes were applied (with one hidden layer) but performed better by adding more layers to it.

4.2.2 Implementing SVM

In implementing SVM, a C value of 1.0 and a regularization (gamma) value of 2.0 was used for all kernels. Tables 4.2 to 4.5 shows results obtained from the implementation of sigmoid, linear Rbf and polynomial hyperplanes.

Using the sigmoid hyper plane, the SVC performance stood at 64 percent accuracy, this is a far cry from aim of this research. This performance clearly showed that this hyper plane does not perform well on this problem, tuning some parameters did not initiate any change. Table 4.4 is a summary from sigmoid plane classifier.

Table 4.4: Result from sigmoid kernel

	Precision	**Recall**	**F1-score**	**support**
Non-churn (0)	0.82	0.64	0.72	1268
Churn (1)	0.41	0.65	0.50	490
Accuracy			0.64	1758
Macro average	0.62	0.64	0.61	1758
Weighted average	0.71	0.64	0.66	1758

Using the linear hyper plane, the SVC experience a gradual increase of 74 percent accuracy. This could be due to the fact the data set is not linearly separable thus abysmal performance. Linear model is simple to train but it gives less accuracy. Table 4.5 exhibits the linear hyper plane classifier implemented.

Table 4.5: Result from linear kernel

	Precision	**Recall**	**F1-score**	**support**
Non-churn (0)	0.91	0.71	0.80	1268
Churn (1)	0.52	0.81	0.64	490
Accuracy			0.74	1758
Macro average	0.72	0.76	0.72	1758
Weighted average	0.80	0.74	0.75	1758

The RBF (Radial basis function) kernel also recorded a high performance of 89 percent accuracy. The data set is not obviously not linearly separable, and the sigmoid kernel was not a good fit for this classification problem. Radial base functions are more costly and takes more time in consuming but gives more accuracy. Table 4.6 below is a pictorial illustration of the above claims.

Table 4.6: Result from RBF kernel

	Precision	**Recall**	**F1-score**	**support**
Non-churn (0)	0.92	0.93	0.92	1268
Churn (1)	0.81	0.80	0.80	490
Accuracy			0.89	1758
Macro average	0.86	0.86	0.86	1758
Weighted average	0.89	0.89	0.89	1758

The polynomial hyper plane out shined the other kernels, in performance, scoring a peak accuracy of 90 percent, but takes longer computational time when compared to the other Kernels. Table 4.7 below summaries results obtained when Polynomial kernel was implemented.

Table 4.7: Result from polynomial kernel

	Precision	**Recall**	**F1-score**	**support**
Non-churn (0)	0.94	0.92	0.93	1268
Churn (1)	0.80	0.85	0.82	490
Accuracy			0.90	1758
Macro average	0.87	0.88	0.88	1758
Weighted average	0.90	0.90	0.90	1758

4.2.3 Implement NN for prediction

After the modeled neural network was trained using the train set prediction was made on the model using the test set and the system scored 73%.

4.2.4 Implementing Adam optimization

Using predefined functions that were used on our neural network model, Adam technique was implemented as well with 10 epochs; 99.7% accuracy was reported from the confusion matrix of this model.

4.3 Discussion

The Table 4.3 below shows a list of all the techniques and their corresponding measure accuracy on the model their result values. The first row displays all machine learning algorithms employed in this thesis. The first column reports accuracy (measured in percentage) for each of four support vector machine kernels that was used, polynomial kernel came out top with 90% accuracy and was closely followed by RBF (radial basis function) kernel that scored 89%. The next column outlines the accuracy obtained from three different optimization techniques that were implemented with Multi-layer perceptron (MLP), Limited memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) technique performed excellently with 91% accuracy, followed by Adaptive moment Estimation (Adam) that scored 91% while Stochastic gradient descent (SGD) had a poor performance with 80% accuracy. The following column portrays the result obtained from employing Neural networks on the data, the model (NN) scored an average performance of 73% (it is worthy of note that no optimization technique was used at this stage). Finally, the last column illustrates the result attained from Adam optimization technique with neural networks with the best overall performance to score 99.7% accuracy.

Table 4.8: Results of all models

SVM		**MLP**		**NN**	**NN with Adam**
Linear Kernel	74%	Adam	91%	73%	99.7%
RBF	89%	SGD	80%		
Polynomial Kernel	90%	LBFGS	91%		
Sigmoid Kernel	64%				

Finally, a brief overview of the entire classification tasks shows that Adam optimization algorithm for NN outperformed the others and SVM (with linear kernel) performed the least as seen from Table 4.8. Deep learning (Deep Neural Networks) works well with in the regime of big data (characterized by huge number of variables), and SVM works well on small multivariate data sets.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

5.1 Conclusion

Data validation was applied to assert the structural integrity of the data and then normalization, generally learning algorithms benefit from standardization of the dataset. To obtain a globally classified data the dataset from Kaggle was split into two parts, the training set- amounted to 75% of the datasets- and then the testing set, composed of 25% of remaining data from the set to achieve a globally cross validated used in all techniques. The classification was executed using all three machine learning techniques by deploying the validated data. Comparison between the techniques was performed, in other to discover accuracies as well as to detect the model that performed best in the classification task.

The results obtained from the system were satisfactory, different output values of the accuracy was obtained by varying parameters, or hyper parameters. The accuracy for all algorithms employed in this research ranged between 64% and 99.7%, the acquired result illustrates effects of the system's performance.

The twentieth century witnessed the advent of big data in the telecoms sector making quite a number of machine learning algorithms (such as decision trees, linear regression, and more) become inapplicable for machine learning. Data from telecom providers are characterized by numerous variables meaning that only robust and sophisticated algorithms, that could handle such data, would be employed to solve machine learning problems. Neural network was applied to predict customers who would churn from the network and with Adam optimization technique it scored the highest point with 99.7 percent accuracy. Neural networks are the preferred choice to solve any machine learning problem (such as image processing or detection, prediction) characterized by big data.

5.2 Future works

Furthermore, practical researches could be carried out on ensemble learning (a combination of multiple algorithms such that the output from one algorithm serves as an input for the next algorithm to be employed) to accurately predict customers who are likely to churn from a network.

REFERENCES

Adhikari, P. R., Vavpetič, A., Kralj, J., Lavrač, N., & Hollmén, J. (2016). Explaining mixture models through semantic pattern mining and banded matrix visualization. *Machine Learning, 105*(1), 3-39.

Ahmed, U., Khan, A., Khan, S. H., Basit, A., Haq, I. U., & Lee, Y. S. (2019). Transfer Learning and Meta Classification Based Deep Churn Prediction System for Telecom Industry. *arXiv preprint arXiv:1901.06091*.

Alspaugh, S., Zokaei, N., Liu, A., Jin, C., & Hearst, M. A. (2018). Futzing and moseying: Interviews with professional data analysts on exploration practices. *IEEE transactions on visualization and computer graphics, 25*(1), 22-31.

Andreas, J., Rohrbach, M., Darrell, T., & Klein, D. (2016). Neural module networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 39-48).

Boobier, T. (2018). *Advanced Analytics and AI: Impact, Implementation, and the Future of Work*. John Wiley & Sons.)

Chang, Z., Lei, L., Zhou, Z., Mao, S., & Ristaniemi, T. (2018). Learn to cache: Machine learning for network edge caching in the big data era. *IEEE Wireless Communications, 25*(3), 28-35.

Dittert, M., Härting, R. C., Reichstein, C., & Bayer, C. (2017, June). A data analytics framework for business in small and medium-sized organizations. *In the Proceedings of the Conference on intelligent decision technologies* (pp. 169-181). Springer, Cham.

Ekaterina, A. (2016). Big data analytics as a marketing tool: The best practices of Russian companies (*Master Thesis), Graduate School of Management,* St. Petersburg, Russia.

Hudaib, A., Dannoun, R., Harfoushi, O., Obiedat, R., & Faris, H. (2015). Hybrid data mining models for predicting customer churn. *International Journal of Communications, Network and System Sciences, 8*(05), 91-95.

Huettmann, F., Craig, E. H., Herrick, K. A., Baltensperger, A. P., Humphries, G. R., Lieske, D. J., ... & Rutzen, I. (2018). Use of machine learning (ML) for predicting and analyzing ecological and 'Presence Only'data: An overview of applications and a good outlook. *In the Proceedings of the Conference on Machine Learning for Ecology and Sustainable Natural Resource Management* (pp. 27-61). Springer, Cham.

Hung, S. Y., Yen, D. C., & Wang, H. Y. (2006). Applying data mining to telecom churn management. *Expert Systems with Applications, 31*(3), 515-524.

Ismail, M. R., Awang, M. K., Rahman, M. N. A., & Makhtar, M. (2015). A multi-layer perceptron approach for customer churn prediction. *International Journal of Multimedia and Ubiquitous Engineering, 10*(7), 213-222.

Jahanzeb, S., & Jabeen, S. (2007). Churn management in the telecom industry of Pakistan: A comparative study of Ufone and Telenor. *Journal of Database Marketing & Customer Strategy Management, 14*(2), 120-129.

Kaggle Inc. (2018). WA_Fn-UseC_-Telco-Customer-Churn.csv (1) [Telcom Customer Churn]. Retrieved from https://www.kaggle.com/blastchar/telco-customer-churn#WA_Fn-UseC_-Telco-Customer-Churn.csv

Karanovic, M., Popovac, M., Sladojevic, S., Arsenovic, M., & Stefanovic, D. (2018). Telecommunication Services Churn Prediction-Deep Learning Approach. *In the Proceedings of the Conference on 26th Telecommunications Forum* (pp. 420-425). IEEE.

Larson, D., & Chang, V. (2016). A review and future direction of agile, business intelligence, analytics and data science. *International Journal of Information Management, 36*(5), 700-710.

Lemley, J., Bazrafkan, S., & Corcoran, P. (2017). Deep Learning for Consumer Devices and Services: Pushing the limits for machine learning, artificial intelligence, and computer vision. *IEEE Consumer Electronics Magazine, 6*(2), 48-56.

Longato, E., Acciaroli, G., Facchinetti, A., Maran, A., & Sparacino, G. (2019). Simple linear support vector machine classifier can distinguish impaired glucose tolerance versus

type 2 diabetes using a reduced set of CGM-based glycemic variability indices. *Journal of diabetes science and technology*, doi.org/10.1177/1932296819838856.

Mahdavinejad, M. S., Rezvan, M., Barekatain, M., Adib, P., Barnaghi, P., & Sheth, A. P. (2018). Machine learning for Internet of Things data analysis: A survey. *Digital Communications and Networks, 4*(3), 161-175.

Manral, L., & Harrigan, K. R. (2018). The logic of demand-side diversification: Evidence from the US telecommunications sector, 1990–1996. *Journal of Business Research, 85,* 127-141.

Mishra, A., & Reddy, U. S. (2017). A novel approach for churn prediction using deep learning. *In the Proceedings of the Conference on Computational Intelligence and Computing Research* (pp. 1-4). IEEE.

Naumann, F., & Herschel, M. (2010). An introduction to duplicate detection. Synthesis Lectures on Data Management, 2(1), 1-87.

Negash, S., & Gray, P. (2008). Business intelligence. In *Handbook on decision support systems 2* (pp. 175-193). Springer, Berlin, Heidelberg.

Noe, R. A., Hollenbeck, J. R., Gerhart, B., & Wright, P. M. (2017). *Human resource management: Gaining a competitive advantage*. New York, NY: McGraw-Hill Education.

O'Neill, N. J. (2019). Standard and Inception-Based Encoder-Decoder Neural Networks for Predicting the Solution Convergence of Design Optimization Algorithms.

Panesar, A. (2019). What Is Machine Learning? Machine Learning and AI for Healthcare, 75–118

Ren, C. R., Hu, Y., & Cui, T. H. (2019). Responses to rival exit: *Product variety, market expansion, and preexisting market structure. Strategic Management Journal, 40*(2), 253-276.

Richards, G., Yeoh, W., Chong, A. Y. L., & Popovič, A. (2019). Business intelligence effectiveness and corporate performance management: an empirical analysis. *Journal of Computer Information Systems, 59*(2), 188-196.

Rodan, A., Faris, H., Alsakran, J., & Al-Kadi, O. (2014). A support vector machine approach for churn prediction in telecom industry. *International journal on information, 17*(8), 3961-3970.

Rodan, A., Fayyoumi, A., Faris, H., Alsakran, J., & Al-Kadi, O. (2015). Negative correlation learning for customer churn prediction: A comparison study. *The Scientific World Journal 15*, 1-7.

Seppälä, J. (2019). Presentation attack detection in automatic speaker verification with deep learning. *Master Thesis, School of Computing*, Kuopio, Finland.

Simões, C. (2016). Characterization of the clients retention in the telecommunications companies. (*Doctoral dissertation)* Management from the NOVA – School of Business and Economics, Carcavelos, Portugal

Vafeiadis, T., Diamantaras, K. I., Sarigiannidis, G., & Chatzisavvas, K. C. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory, 55*, 1-9.

Vercellis, C. (2009). *Business intelligence: data mining and optimization for decision making*. New York: Wiley.

Vishnukumar, H. J., Butting, B., Müller, C., & Sax, E. (2017). Machine learning and deep neural network—Artificial intelligence core for lab and real-world test and validation for ADAS and autonomous vehicles: AI for efficient and quality test and validation. *In the Proceedings of the Conference on Intelligent Systems Conference* (pp. 714-721). IEEE.

Yuan, H. (2016). Labeling large scale social media data using budget-driven One-class SVM classification (Doctoral dissertation, Memorial University of Newfoundland).

Zurada, J. M. (1992). Introduction to artificial neural systems (Vol. 8). St. Paul: West.