Customer Churn Analysis using Deep Learning Models

By

Tan Yi Hao

DEPARTMENT OF COMPUTING AND INFORMATION TECHNOLOGY

TUNKU ABDUL RAHMAN UNIVERSITY OF MANAGEMENT AND TECHNOLOGY

PULAU PINANG

ACADEMIC YEAR

2022/23

Customer Churn Analysis using Deep Learning Models

By

Tan Yi Hao

Supervisor: Dr. Lim Khai Yin

A project report submitted to the
Department of Computing and Information Technology
in partial fulfillment of the requirement for the
Bachelor of Computer Science (Honours)

Department of Computer Science and Information Technology
Tunku Abdul Rahman University of Management and Technology
Pulau Pinang

2022/23

Declaration

The project submitted herewith is a result of my own efforts in totality and in every aspect of the project works. All information that has been obtained from other sources had been fully acknowledged. I understand that any plagiarism, cheating or collusion or any sorts constitutes a breach of Tunku Abdul Rahman University of Management and Technology rules and regulations and would be subjected to disciplinary actions.

_____

Tan Yi Hao

Bachelor of Computer Science (Honours) in Data Science

ID: 21PMR07271

Abstract

Customer churn is a major issue faced by most companies nowadays. It is defined as a group of customers that cancel a product or service over a set period of time. The two types of customer churn are voluntary churner and involuntary churner, and churn analysis are receiving increasing attention in order for businesses to control and identify customer churns. Various models have been built to conduct churn predictions but there is less or no comparison between the models using the same dataset, especially comparison between machine learning and deep learning models. Besides, there were no studies related to churn prediction using a CNN-SVM Hybrid model. In this study, 4 different types of models which are the MLP, SVM, CNN, and hybrid CNN-SVM models are utilized to perform the churn analysis. The main contribution is to develop a hybrid deep learning model that is able to conduct churn predictions and to identify major factors that are affecting the customer churn. Besides, this study also aims to identify which of the models utilized have a better performance in performing churn analysis. The proposed methodology involves the data collection and pre-processing of the "Telco-Customer-Churn" dataset obtained online, and the implementation of the 4 models stated. The confusion matrix of the model's containing the accuracy, precision, recall value, and F1-score is used as the evaluation criteria for the model performance. As a result, the hybrid CNN-SVM model is able to achieve the highest performance with the test accuracy of 95% when using the dataset with the SMOTE-ENN hybrid sampling technique, and the feature "tenure", "OnlineSecurity", "TechSupport", and "Contract" is identified as the major churn factor in the dataset. Comparison between different datasets and cooperation with real world telecommunication company to retrieve real world data is recommended in the future works.

Acknowledgement

Firstly, I would like to thank Tunku Abdul Rahman University of Management and Technology for providing me this precious opportunity to conduct this final year project. I would also like to thank my supervisor, Dr. Lim Khai Yin for providing me guidance and support throughout the progression of this project.

Moreover, I would like to give a moment of appreciation to my teammate, Ng Theng Yang, for providing me with his cooperation and support. I would like to thank him for giving me ideas when I am facing difficulties in certain task for the FYP progression. Without him, I will face major difficulties in completing some of the tasks required.

Lastly, I would like to thank my friends in TARUMT that had given me support and encouragement in completing this whole project.

Table of Contents

Chapter 1: Introduction

Chapter 1: Introduction

In this competitive world, companies tend to reach a saturation state and face fierce competition from their competitors, and customer churn is becoming one of the major issues faced by most companies. Customer churn is defined as a group of customers or subscribers who cancel a service over a set period of time (Agrawal, 2018). In any form of business, the top priority of a business is to gain revenue and profits from its provided products or services, hence customer churn is a serious issue that needs to be tackled by a business. This is because when a customer departs or churns from a company, sales or cross-selling opportunities are also lost at the same time (Dingli, Marmara and Fournier, 2017). This will definitely lead to the loss of revenue and profits to a business. Besides, if a client or consumer leaves a business without offering any advice or recommendations, it may be difficult for the company to make responses and take the necessary corrective action (Dingli, Marmara and Fournier, 2017). The required resources for gaining a new customer is also higher than retaining existing customers (Karanovic *et al.*, 2018).

In order to better control customer churn, companies must completely comprehend their customer's behavioural churn route and the elements affecting customer turnover. There are two types of customer churns, which are voluntary churner and involuntary churners (Townsend and Nilakanta, 2019). Involuntary churner is mostly because of businesses terminating the services provided due to non-payment status, violation of contract, or deception. On the other hand, voluntary churner can also be categorized into two types, which are deliberate churn, and incidental churn. Deliberate churn is caused due to customers not being satisfied with the products or services provided by the business in terms of economic factors (price sensitivity), technology factor (more advanced technologies are provided by competitors), poor customer service, or other factors that cause inconveniences to the customers. Incidental churn is caused mainly by the sudden changes in the daily lives of the customers, such as changes in financial status, or geographical changes (Townsend and Nilakanta, 2019).

1.1 Problem Statement

In order to control and identify customer churn, churn analysis or prediction is receiving increasing attention over the past time in the marketing and management literature (Tsai and

Lu, 2009). The main goal of conducting churn analysis is to classify the customer of a company into churner and non-churner (Mishra and Reddy, 2017). It is a method to determine profiles of the company's existing customers, analyze factors causing customer churn, and predicting customer churn (Çelik and Osmanoğlu, 2019). This paper aims to conduct churn analysis using the dataset retrieved online. The major churn factors of the data are also identified throughout this research.

In order to conduct churn analysis, various statistical and machine learning models such as Random Forest (RF), Support Vector Machine (SVM) had been deployed in order to create a more accurate prediction model (Tsai and Lu, 2009). Besides from traditional machine learning method, various deep learning models had also been utilized to perform the churn prediction. Among the deep learning models, Multilayer Perceptron (MLP) Neural Network and Convolutional Neural Network (CNN) are among the most used models to perform churn analysis. A previous study conducted by Ismail *et al.*, 2015 had investigated the application of MLP Neural Network in churn prediction for a telecommunication company. The result had proven that MLP Neural Network is better over other model with a prediction accuracy of 91.28%. Another study conducted by Amuda and Adeyemo, 2019 had also used MLP Neural Network to perform churn prediction in a financial institution. Accuracy rates of 97.53% had been achieved using python. Hence, it is suitable to say that MLP Neural Network is also a suitable model to conduct churn analysis in this paper.

As for the CNN model, even though theoretically CNN model is more suitable for dealing with image data (Albawi, Mohammed and Alzawi, 2017), however, in a study conducted by Mirjana Karanovic, it is shown that CNN can also be used in one-dimensional imbalanced data for churn prediction. The results show that CNN is able to achieve an accuracy of 98.85% in performing the customer churn prediction (Karanovic *et al.*, 2018). Thus, it is proven that CNN is capable of conducting churn predictions. However, none or only a few direct comparisons had been done towards the performance of both models on the same dataset. Hence, there is a gap between which algorithm is more suitable to perform the churn analysis, and it is beneficial to find out which models perform better than the others. Besides, since there is no previous study conducted that combines the CNN and SVM model, a hybrid model which uses CNN as an automatic feature extractor and SVM as a binary classifier is proposed in order to determine whether this hybrid model is suitable for tasks related to churn prediction. Since it is stated in Xia and Jin, 2008, that SVM that uses structural risk minimization method was having a better

results as compared to other models that applied the empirical risk minimization method, it is anticipated that the automated feature extractor of the CNN model combined with the SVM model will produce a better model performance.

1.2 Research Objectives

The objectives of this study are:

1. **To develop deep learning models by combining CNN and SVM model that are capable of conducting churn prediction**

   As stated above, churn prediction is receiving increasing attention in recent times. Hence, this research aims to build several different models, which are MLP, CNN, and CNN-SVM hybrid models, to conduct churn prediction using datasets available online. By utilizing these models, the following sub-objectives will also be conducted:

   i. To identify potential factors leading to customer churns

   This research also aims to identify the major factors causing customer churns. This enables the company to create suitable marketing strategies using the factors identified by the churn prediction model and fulfil more individual and unique needs of their customers (Ahn *et al.*, 2020). Hence, the overall customer satisfaction can be increased, and the probability of potential churn will also decrease.

   ii. To compare the efficiency of the proposed model in conducting churn analysis

   This research also aims to identify which deep learning model is more suitable for conducting churn analysis. Many of the previous studies conducted had proven that deep learning models are suitable for churn prediction, however none or only a few of them had made comparison between the performance of the deep learning model. This paper aims to fill in this gap by comparing the performance of the models developed in performing churn analysis.

1.3 Research Scope

In this study, customer churn analysis is conducted by implementing primarily deep learning models. With the increase in processing capacity and new data source over the past few years, deep learning approaches have taken advantage of previously unused features such as social

network or textual information to improve the performance of the churn prediction model (Mena *et al.*, 2019). Besides, deep learning approaches can also solve some of the issues faced by traditional machine learning approaches. For example, in the traditional machine learning approach, a major problem is that the manual feature engineering process will be very tedious and time consuming, and it will often require a domain expert to conduct the feature engineering. Another major problem is that the models built by traditional machine learning approaches are typically tailored to a specific dataset. Hence, the feature engineering process will need to be performed for every different dataset used (Umayaparvathi and Iyakutti, 2017). All of these issues can be solved by using the deep learning approach as it is able to come up with good features and representation of the input data automatically (Umayaparvathi and Iyakutti, 2017).

In this study, besides from a SVM model, two of the deep learning algorithms are used to build the churn prediction model. The algorithms that are applied in this study are the MLP and CNN model. Even though theoretically CNN model is more suitable for dealing with image data (Albawi, Mohammed and Alzawi, 2017), however, the results in Karanovic *et al.*, 2018 show that CNN is also capable of conducting churn predictions. Hence, in the proposed paper, comparison between the SVM, MLP Neural Network and CNN model is also proposed to identify which algorithm is more suitable for conducting the churn analysis.

Besides the standard model, a hybrid model which combines both CNN and SVM is proposed in this study. CNN will act as an automatic feature extractor while SVM will act as a binary classifier for the churn prediction.

The research method that is utilized for this project is quantitative research. Numerical data will be collected and analyzed to perform churn prediction and generalize results for research. The dataset used in this research will be obtained online through the website “kaggle”.

1.4 Research Milestone

Table 1.1 Research Milestone

Milestone	Deadline
Introduction	30/5/2022
Literature Review	15/6/2022
Methodology	30/6/2022
Data Understanding	23/7/2022
Data Pre-processing	28/8/2022
Modelling (MLP Neural Network)	20/9/2022
Modelling (SVM)	20/10/2022
Modelling (CNN)	12/11/2022
Modelling (CNN-SVM)	25/11/2022
Result Analysis & Model Evaluation	15/12/2022
Discussion, Conclusion & Future Work	26/12/2022
Total Duration: 25/5/2022 – 26/12/2022	

1.5 Thesis Outline

A total of four chapters was completed during project 1. In Chapter 1, a brief introduction about churn analysis is given. The problem statement, objective, research scope, and research milestone are also stated.

In Chapter 2, literature review of the various machine learning and deep learning algorithms used to conduct churn analysis is provided. A conclusion justifying the algorithms used in this research is also provided.

Chapter 3 discuss the methodology of this research. The methodology used for data collection, data pre-processing, modelling and performance evaluation is discussed in this chapter.

Chapter 4 discuss the theoretical background of the chosen algorithm to better understand how those algorithms work.

Chapter 5 provides the performance comparison between the proposed algorithms and the various methods utilized in this study.

In Chapter 6, discussions concerning the result obtained in chapter 5 are conducted.

Lastly, in Chapter 7, the conclusion of this research is provided. The limitations of this study and the future works recommended are also provided.

Chapter 2: Research Background

Chapter 2 Research Background

In this chapter, past studies about churn analysis using machine learning or deep learning algorithms is discussed. This chapter is divided into two parts. The first part contains the literature review of the previous studies. It provides a brief explanation of what past researchers have done and the result of their research. On the other hand, the second part contains the discussion and conclusion.

2.1 Literature Review

2.1.1 K Nearest Neighbor (KNN)

In (Sjarif *et al.*, 2019), a KNN model was utilized to conduct churn prediction on the "Telco-Customer-Churn" dataset. Pearson Correlation Coefficient method had been used for the selection of attributes. The dataset was split into training and testing data by using a 70:30 ratio. As the number of neighbors (k) in the model can be modified to achieve higher performance, different values of k ranging from 1 to 20 were used to compare the performance of the model. Two types of evaluation methods had been used to compare the models, which are accuracy and confusion matrix. At the end, the result showed that the best training accuracy of 80.45% was achieved when the value of k is 18, while the best testing accuracy of 97.78% was achieved when the value of k is 1. The study also included comparison of results of other algorithms to the proposed model which are Random Forest and Support Vector Machine (SVM), and it was also shown that the KNN algorithm outperforms other models where both the Random Forest and SVM model achieved an accuracy of below 80%.

In (J.Alsakran, 2019), a comparison study between Decision Tree and KNN was conducted to find out which model has the highest performance in performing churn prediction. Both models were fed with the same input data with the same training and testing ratios and the input data was processed through filters in order to reduce noise and undesired data such as outliers. The dataset used in the experiment belongs to a telecommunications company, which consists of 3333 samples. 95% of the sample was split into both training and testing data using a 60:40 ratio. As for the experiment setup of the KNN model, the number of neighbors was configured to k = 5 after conducting several trials, and mixed Euclidean distance measures had been used to calculate the distance between the data points. The performance of the model was evaluated through various criteria such as the accuracy, precision, recall value, F-Measure, AUC, and lift measure. In the result of the experiment, it was shown that the decision tree model outperforms

the KNN model with the accuracy of 93% compared to 87%, and the largest difference was with the F1 score with 33% and 73% for Decision Tree and KNN respectively. It was stated that this was caused due to the recall measure being very low for the KNN model, and it was caused by the number of K neighbors in the model. However, while trying to improve the recall by decreasing the number of K in the model, the precision value will also be affected and decrease at the same ratio, resulting in similar results for the F1 score. In the AUC graph, it was also shown that the Decision Tree model has higher true negative rates than the KNN model. By viewing the results stated, it can be said that Decision Tree is found to be more efficient that KNN model, however, the results in the confusion matrix also shows that the KNN model was able to predict the true positives rate more accurately the the Decision Tree model, which means the KNN model was able to predict the real churning customer better that the Decision Tree model. Hence, it is stated that future works should measure other values to better prove the efficiency of the models.

Lastly, in (Joshi and Gupta, 2019), a study to predict customer churn in Telecom Industry was conducted by using Centroid Oversampling method and KNN classifier. The reason oversampling method was used was to handle class-imbalance problems. In the paper, the Centroid Oversampling method was proposed to compare with one of the oversampling techniques, which is Synthetic Minority Oversampling Technique (SMOTE). This is because one of the major concerns of SMOTE is that as SMOKE produces new minority data points based on the nearest neighbors, classes with frequent samples will overrule neighbors of testing instances in-spite of the distance metrics. Centroid Oversampling method was proposed to resolve this issue by generating synthetic samples through identifying the neighbors and compute the centroid of these data. Hence, synthetic samples can be distributed uniformly thus reducing the probability of choosing outliers in the data space. The original dataset was oversampled by 100% through this method. In the result, comparison was made between three types of models, which is the KNN model, KNN model which applies SMOTE oversampling method, and KNN model which applies Centroid oversampling method. Various dataset and various numbers of k had also been applied to compare the performance of these models on different datasets. As a result, it was shown that the classification accuracy for KNN with Centroid Oversampling method outperform other models.

2.1.2 Logistic Regression

In (Jain, Khunteta and Srivastava, 2020), churn prediction in telecommunication had been conducted using Logistic Regression and Logit Boost algorithm. The experiment was conducted by using the dataset provided by an American telecommunication company. The model was also measured using many performance measures such as Kappa Statistics, Mean Absoulte Error, Root Mean Square Error, Relative Absolute Error, and the confusion matrix. In this experiment, the results showed that both the Logistic Regression and Logit Boost model can achieve similarly good results. Both models had achieved an accuracy of 85.2385% and 85.1875% respectively. The Logistic Regression model can better classify the customers that will not churn, however, the Logit Boost model can better classify the customers that would have churn. In future works, a hybrid model was proposed to embed the characteristics of different algorithms to further enhance the model performance.

In (Peng, 2014), Cluster Stratified Sampling logistic regression model (CSS-LRM) was applied to predict customer churn. The reason sampling method was used is because in actual application, the datasets will usually be imbalanced, and there will be larger gaps between the positive instance and negative instance. Hence, if Logistic Regression is applied without any sampling method, the performance of the model will often not be good as the data have serious data sparse problems. Hence, the paper utilized the stratified sampling method to resolve this issue by randomly extracting examples from both the positive and negative instances then merge the training samples for parameter estimations. In the experiment, two popular datasets for churn analysis (Churn and Orange) were used to conduct the prediction. The churn set represents a dataset that has no missing values and less data pre-processing steps are required for this dataset. However, in the Orange dataset, there are many missing values, dimensions and complex information. Hence, a large amount of pre-processing work needs to be performed for this dataset. Two models had been applied to make comparison between the performance between the models in predicting both balance and imbalance datasets. The first method was by utilising SVM with common RBF kernel function, while the second method was by using CSS-LRM with parameter compensation. At the end, the result showed that both SVM and CSS-LRM can achieve good performance in the Churn dataset, which is a balanced dataset. However, in the Orange dataset which is an imbalanced dataset, the performance of SVM was worse than the CSS-LRM model with ROC values of 0.5692 and 0.7354 respectively.

2.1.3 Decision Tree

In (Luo, Shao and Liu, 2007), customer churn prediction had been performed in personal handyphone system service (PHSS) by using Decision Tree algorithm. In the paper, the aim was to overcome the limitations of the lack of information of the customers in this sector and to build an efficient model that is able to predict the customer churn. In the informations provided, there are only three kinds of information that can be used to build the model, which are "Frequency of use (FOU)", "Sphere of influence (SOI)", and "Minutes of use (MOU)". Hence, in order to build an effective model, three types of experimentation had been performed to improve the ability of the decision tree model to perform the analysis, which were to change the sub-periods during data pre-treatment process, to change the churn model misclassification cost, and to change the sampling methods used for the training datasets. As a result, the optimal parameter to train a decision tree model to predict customer churn of PHSS was identified, which is having 10 days of the duration of sub-period, having a misclassification cost of 1:5, and utilizing random sample methods for the training data.

In (Pejić Bach, Pivar and Jaković, 2021), a framework of combining both clustering and classification to perform churn management was proposed using k-means cluster analysis and decision tree. In the proposed framework, cluster analysis was conducted to group the homogenous customers into the same cluster and compare the groups by the churn ratio. On the other hand, the churn prediction was performed to the groups that have the highest churn rate. On this stage, feature selection was conducted using chi-square statistics. A decision tree model was then applied to the groups identified to have the highest churn rate with the feature selected. As a result, it is found that major factors of customer churns in the telecommunication sector is due to bad initial experiences and low satisfaction of service provided. In the study, it is also shown that churn mostly occurs with customers who are using trial or prepaid accounts.

2.1.4 Support Vector Machine (SVM)

In (Farquad, Ravi and Raju, 2014), a hybrid approach was proposed to perform churn prediction using SVM and Naive Bayes Tree (NBTree). As the dataset used in this study was highly imbalanced with 93.24% of loyal customers and only 6.76% of churned customers, various sampling methods had also been applied to resolve this issue. The proposed model consists of three phases, which was to reduce the feature set using SVM-recursive feature elimination (SVM-RFE) method, extract the support vectors from the SVM model, and generate rules using the NBTree model in the final phase. In the second phase during the

support vectors extraction, three datasets were generated with “Case-SP" being the dataset with actual target values of support vectors being replaced by the predicted target values, “Case-SA” being the dataset with the actual target values of support vectors, and “Case-P” being the dataset with the actual target value of the training instances being replaced by the predicted target values. As a result, it is shown that the best model is SVM + NBTree model using “Case-SP”" dataset with reduced feature balanced by SMOTE sampling technique, which achieved the highest sensitivity of 91.85%

In (BRANDUSOIU and TODEREAN, 2013), SVM models with different kernel functions had been built to perform churn predictions in the telecommunications sector. In the proposed models, four of the common kernels, which are Radial Basis Function kernel (RBF), Linear kernel (LIN), Polynomial kernel (POL), and Sigmoid kernel (SIG), had been implemented to perform the prediction. The dataset used in the study was split into training and testing data with an 80:20 ratio. Confusion matrix had been applied to make comparison between the models with different kernels. As a result, it is shown that the model that utilizes the polynomial kernel function has the highest performance of an overall accuracy of 88.56%, while the model with sigmoid kernel function has the lowest performance of an overall accuracy of 70.53%.

In (Xia and Jin, 2008), it states that the methods used in previous research such as decision tree or logistic regression can only be used to analyze qualitative and continuous data, and make interpretation. However, it doesn’t guarantee performance of the model on large scale, high-dimensionality, nonlinearity, and time series data. Even though this problem can be solved using artificial intelligence methods such as ANN or Self Organizing Map (SOM), it will cause low generalization ability and fuzzy construction of the models due to the use of empirical risk minimization. In order to solve these problems, the paper proposed a SVM model with structural risk minimization and the new model evaluation standard for churn prediction. Two datasets were used in this paper, which is the dataset from machine learning UCI database of University of California, and a home telecommunication carrier. Both of the datasets were pre-processed due to it having anomalies, and are being imbalanced. As for the model, the SVM model was compared with other models such as backpropagation ANN, decision tree C4.5, logistic regression, and naive bayesian classifier. As a result, it is shown that the SVM model was able to achieve the best results in all evaluations, which are the accuracy rate, hit rate, coverage rate, and lift coefficient, for both datasets. It concludes that SVM has good prediction capabilities for churn prediction. However, solutions to problems such as selecting the optimal kernel function and parameters, and ways to weigh customer samples can serve as a future

research direction. It also states that since the data in the banking industries are similar to the telecommunication industries, SVM model can also be used for churn prediction in the banking sector.

2.1.5 Random Forest (RF)

In (Xie *et al.*, 2009), it was stated that data imbalance is a serious problem in conducting churn prediction as the number of customers that actually churn are significantly lesser than those who don't churn in the data given. Although various different methods had been used to address this problem, the results given are still proven to be unsatisfactory. Hence, the paper proposed an improved balanced random forest (IBRF) model to conduct churn prediction. It aims to investigate the effectiveness of random forest in conducting churn prediction as well as integrate various sampling techniques and cost sensitive learnings into the model in order to achieve higher performance. It introduced "interval variables", which alter the class distributions and cause heavier penalties on minority class misclassification. To achieve higher noise tolerance, the interval variables determine the sample distributions in different iterations, thus maintaining the randomicity of the sample selection. This allows the ineffective and unstable weak classifiers to be able to be trained based on a more balanced dataset and appropriate distinguishing measure. The IBRF was built by combining the balanced random forest and weighted random forests. Thus, it was able to achieve the benefits of noise tolerance and efficiently deal with imbalance data through balanced random forest, and at the same time, the cost-sensitive learning used weighted random forest helps on the classifier produced by the decision-tree learning methods. For the evaluation criteria, lift curve and top-decile lift were used as a performance measure for the model. The IBRF model was compared with other classifications models, which are ANN, and decision tree. As a result, it is found that the IBRF model achieved significantly higher performance than both other models with an accuracy rate of 93.2%. The IBRF was also compared with other random forest models, which are the balanced and weighted models, and the result shows that in the lift curve, the discriminability for the IBRF model is significantly higher than the other models. Hence, it is proved that the IBRF has a higher performance than other models. This paper also states that future research direction should focus on improving the effectiveness and generalization ability of the IBRF model.

In (Kaur and Kaur, 2020), the usage of various churn prediction models were focused on the banking sector. This is because the banking industry is facing challenges to retain its customers

due to various reasons, such as better financial services by competitors, branch locations, digital tools quality, and lower interest rates. In the paper, random forest, logistic regression, decision tree, KNN were applied. As the dataset used contains missing values and imbalance classes, data pre-processing was conducted to prepare the data for training and testing. Exploratory Data Analysis (EDA) was used to provide maximum insights to the dataset. Bivariate Analysis was also used to find out the relationship between the variables of the dataset. The feature selection process was also performed using chi-square test, and the top 9 features were used for the baseline model building and evaluation. As for the performance evaluation, the accuracy, recall, precision, and AUC and ROC curve were used to evaluate the prediction model. The performance comparison was made through different conditions, which are classifiers with stratified sampling, without stratified sampling, and 8-fold cross-validation. As a result, it is shown that in all cases, the model with stratified sampling and cross validations performs better among other classifiers. On the other hand, the random forest model achieved the highest performance with the training data that contains all features. Ensembling methods were also utilized using voting and averaging techniques to improve the overall performance of the model. As a result, it is shown that the random forest model still achieved the highest result of 85.22% after the ensembling process. Lastly, it states that future works should focus on more advanced machine learning techniques such as neural networks as well as more advanced ensembling techniques such as boosting, bagging to improve the model performance.

2.1.6 Multilayer Perceptron Neural Network (MLP-NN)

In (Agrawal, 2018), prediction of churn on a telecommunication dataset was performed through a deep learning approach. The aim was to build a churn prediction model using a multi-layered neural network that can predict the possibility of customer churn as well as the major churn factors in the dataset. The dataset used in the study was the "IBM Watson Telco Customer Churn Data Set", which has a total of 7043 user data. By using a deep learning approach, the challenges faced during feature extraction could be prevented as deep learning models are able to learn to focus on the important features in the dataset by themselves. However, One Hot Categorical Encoding technique was used during data pre-processing to convert textual data into numerical data as deep learning models do not work well with textual data. As for the neural network model, the number of neurons for the first layer had been set to 16 to achieve optimized results. Dropout layer had been added once every 2 layers to prevent the model from being too complex and causing overfitting. At the end, a 5-layered ANN had been built for model training and prediction. The model was trained with 100 epochs and the batch size for

each epoch was 10. The model was implemented using Keras neural networks API, a Python deep learning library. At the end of the experiment, the result shows that the model was able to achieve an accuracy of 80.03%. Besides, major factors to prevent churns are also identified through correlation analysis, which was "1-year tenure", "Month-to-month contracts", and "Fibre optic internet services".

In another study conducted by (Tsai and Lu, 2009), two hybrid neural network models had been built to conduct churn prediction, which are ANN combined with ANN (ANN + ANN), and SOM combined with ANN (SOM + ANN). The first techniques in both models were used to filter unrepresentative training data, while the second technique were used to create the prediction models based on the filtered representative data. For the study, a CRM dataset was provided by an American telecom company, which consists of 51,306 data, having 34,761 churners and 16,545 non-churners. In the experiment, comparison had been made between the baseline model and the two hybrid models. Different learning epochs and hidden layer nodes were also used to obtain the optimized baseline model. To evaluate the performance of the models, three types of testing sets, which were the general testing set and two fuzzy testing sets based on the filtered data by ANN and SOM had been used, and the experimental results are evaluated in terms of the prediction accuracy and the Type 1 and 2 errors. Based on the results, it is shown that both the hybrid models outperform the baseline model using general testing sets, however the SOM + ANN model does not perform better than the base model when using fuzzy testing sets. Hence, it is safe to say that hybrid models will perform better than the baseline model, and the ANN + ANN model has a more stable performance compared to the SOM + ANN model.

Lastly, in a study conducted by (Umayaparvathi and Iyakutti, 2017), three deep neural network architectures were used to perform automated feature selection churn prediction. The aim is to show that deep-learning based models can perform as good as traditional machine learning models without manual feature selections. The three deep neural network architectures used were small Feedforward Neural Networks (SFNN), large Feedforward Neural Networks (LFNN), and Convolutional Neural Network (CNN). Two different real telecom datasets had been used to evaluate the models. For the "CrowdAnalytix" dataset, LFNN achieves the highest accuracy of 93.1% while SFNN performs worse than the baseline model which is 91.24% and 92.77% respectively. On the other hand, for the "Cell2Cell" dataset, LFNN and CNN achieved the same accuracy of 71.66% while SFNN also performs worse than the baseline model which are 64.75% and 71.27% respectively. Hence, the results show that deep learning models do

perform equally efficiently as traditional machine learning models like random forest and SVM.

2.1.7 Convolutional Neural Network (CNN)

In (Karanovic *et al.*, 2018), CNN had been used to conduct churn prediction on a dataset provided by the telecom company – Orange. The main contribution of this paper is to prove that CNN is also able to be applied on one-dimensional data, as well as able to be used to perform churn prediction. As the dataset is an imbalanced dataset, large amounts of data pre-processing need to be done to ensure better model performance. Features with more that 30% missing values were excluded from the classification process, and mean value was used for numerical data and most frequent value for categorical data for the missing value of the features below the threshold value. Lasso regression was utilized to perform the feature selection and regularization to enhance the prediction accuracy and interpretability. Lastly, manual feature engineering had also been conducted, leading to a total number of 53 features left for the training process. As for the model, in the research, three CNN models were built, with each one being more complex than the previous one. With more complex models, the computation time increased significantly, however, the performance of the model just slightly improved. Hence, only the first proposed model was chosen for performance evaluation. In order to evaluate the model performance, the model sensitivity, specificity, accuracy, and AUC had been used. Overall, the model can achieve an accuracy of 98.85%, which is a high performance. Thus, it is proven that CNN can perform churn predictions and also achieve similar or better accuracy than other models.

In another study by (Mishra and Reddy, 2017), CNN was also implemented for churn prediction using another dataset. The dataset was divided into training and testing data by a 60:40 ratio. The CNN model was built using the R studio tool and the performance of the model was then evaluated using the model accuracy, error rate, precision, recall, and F1-score. In the result, it is shown that the CNN model is able to achieve an accuracy of 86.85%, error rate of 13.15%, precision score of 91.08%, recall value of 93.08%, and F1-score of 92.06%, which indicates a good model performance.

2.1.8 Long Short-term Memory (LSTM)

In (Mena *et al.*, 2019), the LSTM model was used to determine the performance of the model with RFM variables in comparison with other standard models. In the paper, a regularized logistic regression model was used as a baseline model in order to make the comparison with

the LSTM model. Nested cross-validation procedure with three outer folds, and four inner folds was conducted to tune the parameters of the models. In order to prevent overfitting, the training sample was also divided into k-folds for both inner and outer fold. AUC curve, top-decile lift, and expected maximum profit (EMPC) were used to evaluate the model performance. AUC curve acts as the performance measure of the classifier, while with top-decile lift, the performance of the model can be evaluated within the top-ten percent highest probability of customer attrition, which will be useful for targeted marketing. On the other hand, through the expected maximum profit, the model can be evaluated from a marketing perspective. As a result, it is shown that both the top-decile lift, and EMPC measures of the LSTM model are higher than the baseline model. The result showed important implications of churn modeling in the financial industries since RFM data is readily available in this sector. This study also stated that it may be beneficial to incorporate various types of dynamic behavioral data in churn prediction models through deep learning methods, and it stated that this is an open area for future studies.

In (Alboukaey, Joukhadar and Ghneim, 2020), it was stated that most of the previous study of churn prediction only looked at monthly level behavior. However, it also states that the model performance will be affected if it ignores the changes of customer behaviours over days of month. Besides, it also states that it will be too late to retain customers who decided to leave during the start of the month as those customers will only be classified as churners during the next month. Hence, a daily churn-prediction model was proposed in the paper to address these issues. Four models were developed in the paper. Two of the models, which are the RFM-based model and statistics-based model, used the features extracted from the multivariate time series. The other two models, which are the LSTM-based model and CNN-based model, depended on the deep learning technique of automated feature extraction. As for comparison, the models built were compared to various other approaches by previous works, in which most of them are models for monthly churn prediction. The analysis of the performance of both types of churn models, namely daily and monthly, were also provided in the future time window in order to prove the benefits provided by the daily churn prediction model in detecting potential churn earlier. As for the model performance evaluation, AUC curve, logarithmic likelihood of the classification (log loss), F1-score, top-decile lift, and EMPC were used. As a result, the finding shows that daily models are more efficient in detecting churners than monthly models. This finding is very important as it helps in increasing the efficiency of retention marketing campaigns of a company. The result also shows the LSTM model outperforms the CNN model,

however both models are equal to the RFM-based models in terms of prediction performances, with the statistics-based models being outperformed by all other models. The paper also suggested that future works should consider features such as dynamic social features. Besides, it also suggested addressing the issue of interpretability and retention effectiveness. It suggested that future works should learn the customer response towards retention interventions by adding dummy variables that denote customer responses to the models.

In (Bayrak *et al.*, 2021), the fast food industries had been stated to be an extremely competitive environment due to various brands offering different kinds of products at different prices. Therefore, it is stated that customers in the fast food industry are not particularly loyal, which is a difficulty for conducting churn prediction. Besides, it is also stated that the order frequency of the customers in this industry vary, which causes another difficulty faced to conduct churn prediction. These difficulties cause study related to churn prediction of the fast food industry to be infrequent. Hence, in the paper, a RNN model with long short-term memory units was proposed to deal with the sequential patterns for the customer data that are changing over time. Different from other studies, the paper labels the data for different churn phases when a churn event is occurring, so that the churn status of the customer can be predicted before they actually churned. The churn period was also calculated independently for each customer with different approaches. These provide more realistic results since the regularity period and day-related features were calculated independently. As for the classification methods, besides from the LSTM model, RF, SVM, and MLP models were also built with non-sequential data for comparison purposes. The LSTM model was built with a bi-directional approach, in which the hidden node is set to 123, and the batch size is set to be 64. Dropout layer was also added into the model to avoid overfitting. Sigmoid activation function was used as the activation method of the model, and Adam was chosen as the optimization parameter. As a result, it is shown that the LSTM model significantly outperformed other models with non-sequential data in both the experiment for all-cumulated data and period-cumulated data. It reaches the F1 score of 77.90% for all-cumulated data, and 76.19% for period-cumulated data.

2.2 Discussion and Conclusion of Research Background

In summary, a total of 20 papers related to churn analysis using different methods had been reviewed. It is shown that imbalance datasets (Kaggle, n.d.) will be very likely to affect the model performance in conducting churn predictions as some of the papers had used various sampling methods to resolve this issue to ensure higher model performance.

Besides, it is also shown that compared to traditional machine learning models, deep learning models are also able to achieve similar or even better results than the machine learning model. For example, it is stated Karanovic *et al.*, 2018, that CNN model is able to achieve the highest performance compared to traditional machine learning model. Besides, automated feature selection is also a strength of deep-learning methods.

Furthermore, in studies related to LSTM models, it shows that LSTM models are commonly used for sequential or time-series data, in which the dataset selected for this study does not have those properties. Hence, LSTM models is not be suitable to be utilized in this study. This is because the dataset used in this study contains mostly textual data instead of continuous data, in which LSTM does not work well with.

In conclusion, the algorithms used in this paper will be Multilayer Perceptron Neural Network (MLP-NN) and Convolutional Neural Network (CNN), in which both are deep learning models. Comparison between both models will also be conducted to find out which neural networks performs better in churn analysis. Various sampling methods is utilised as the dataset chosen is an imbalance dataset. In addition, a hybrid approach combining CNN and SVM is also proposed. This is because since it is stated in Xia and Jin, 2008 that SVM which applies the structural risk minimization method can conduct churn better that empirical risk minimization method, it may be suitable to find out whether the automated feature extraction used by the CNN model can be combined with the SVM model may produce better model performance, as it is shown to have effective results in other applications **architecture** (Ahlawat and Choudhary, 2020), and none of the previous studies related to churn prediction had tried to apply this approach.

Chapter 3: Methodology and Requirement Analysis

Chapter 3 Methodology and Requirement Analysis

In this chapter, the methodology used for the data preparation, data pre-processing, modelling, and model performance evaluation are discussed. It provides, break down and discuss the general framework of the entire churn analysis process (Figure 3.1).

3.1 General Framework of Churn Analysis

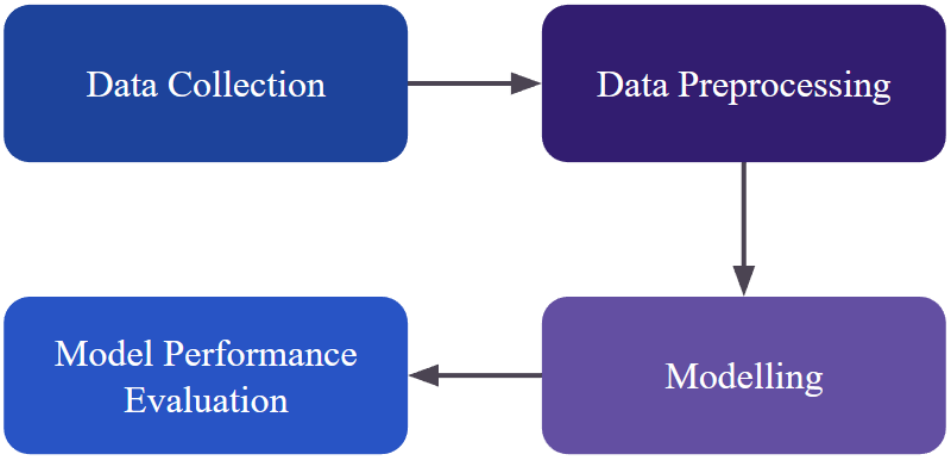


Figure 3.1 Flowchart of General Framework for Churn Analysis

3.1.1 Data Collection

The dataset used in this experiment is a dataset available online publicly called the "Telco-Customer-Churn" dataset (Kaggle, n.d.). It is widely used for research on churn analysis and includes information such as the customer churn, services the customer signed up to, customer account information, and customer demographic information. There is a total of 7043 rows of data with 21 columns in the dataset. In the dataset, there are a total of 5174 data for non-churner and 1869 data for churner (Figure 3.2). The detail of the attributes in the dataset are listed below in table 3.1:

Table 3.1 Dataset Attributes

Attributes	Data Type
CustomerID	object
gender	object
SeniorCitizen	int64
Partner	object
Dependents	object

tenure	int64
PhoneService	object
MultipleLines	object
InternetService	object
OnlineSecurity	object
OnlineBackup	object
DeviceProtection	object
TechSupport	object
StreamingTV	object
StreamingMovies	object
Contract	object
PaperlessBilling	object
PaymentMethod	object
MonthlyCharges	float64
TotalCharges	object
Churn	object

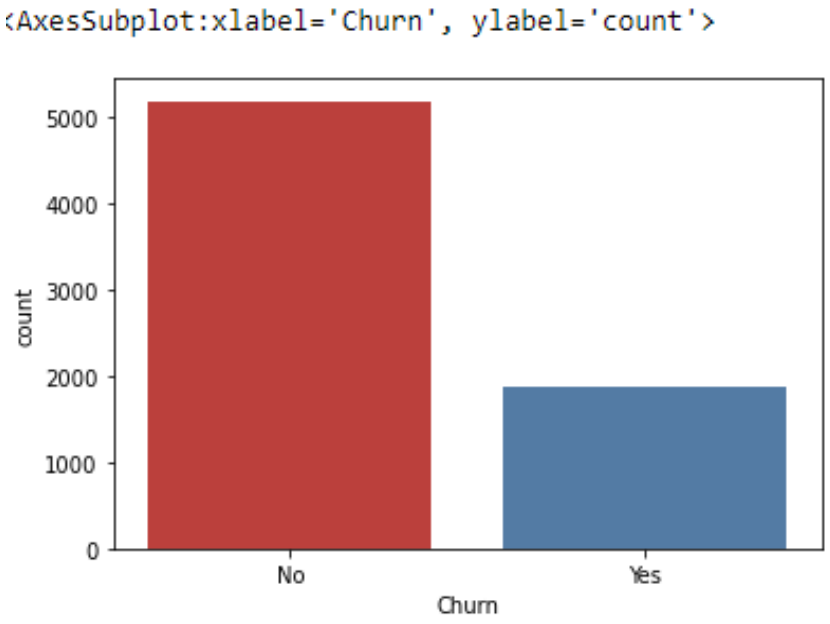


Figure 3.2 Barchart for count of churners and non-churners

3.1.2 Data Pre-processing

After the raw data has been obtained, data pre-processing is performed to ensure that the data is suitable to perform the analysis. Figure 3.3 shows the pre-processing process that are applied to the dataset:

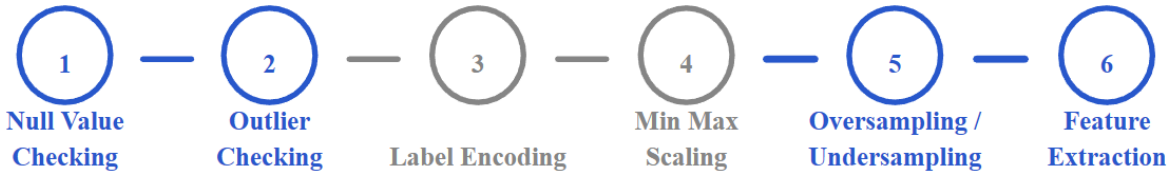


Figure 3.3 Pre-processing applied to the dataset

Firstly, null value and outlier checking is conducted to clean the dataset. After null value and outlier checking is completed, label encoding is conducted as most of the attributes of the datasets are in textual format and some of the models used in this study do not work well with textual data. Min max scaling is also conducted to scale the data to the range of (0,1).

Oversampling and undersampling technique are also used to deal with the class imbalance problem of the dataset. In this study, the two of the sampling techniques are used, which are Synthetic Minority Oversampling Technique (SMOTE), and SMOTE + Edited Nearest Neighbor (ENN) hybrid sampling technique. Lastly, three different types of feature extraction technique are used in this study, which are the random forest feature importance, principal component analysis (PCA), and correlation analysis. Comparison of the model performance using different sampling and feature extraction method are conducted using the MLP model with the same parameter setting.

3.1.3 Modelling

During this phase, modelling for the 4 models are conducted. The general process for building the MLP and SVM model are the same, with the CNN and hybrid model having slightly different processes. The flowchart below shows the general process of building the 4 models **(Figure 3.4, Figure 3.5, Figure 3.6)**:

MLP and SVM

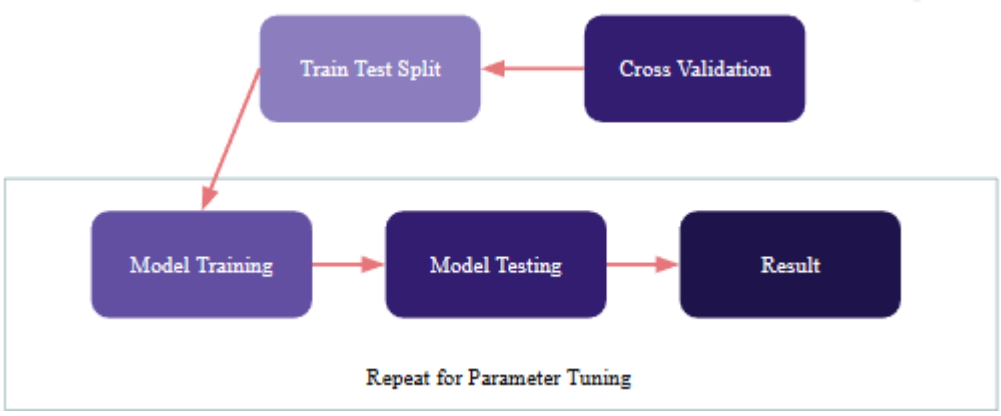


Figure 3.4 Modelling Process of MLP and SVM Model

During this phase, 5-fold cross validation is performed to ensure that the model can perform well when using different training and testing data since the model performance can be different depending on the training data. When building the model, the dataset is split into 80% as a training dataset for the model, and 20% for testing purposes. After splitting the data, the training data is fit into both the MLP and SVM model and model testing is conducted to evaluate the model performances. The modelling process is then repeated to conduct parameter tuning in order to obtain the optimal parameter setting to get the highest performance possible.

CNN

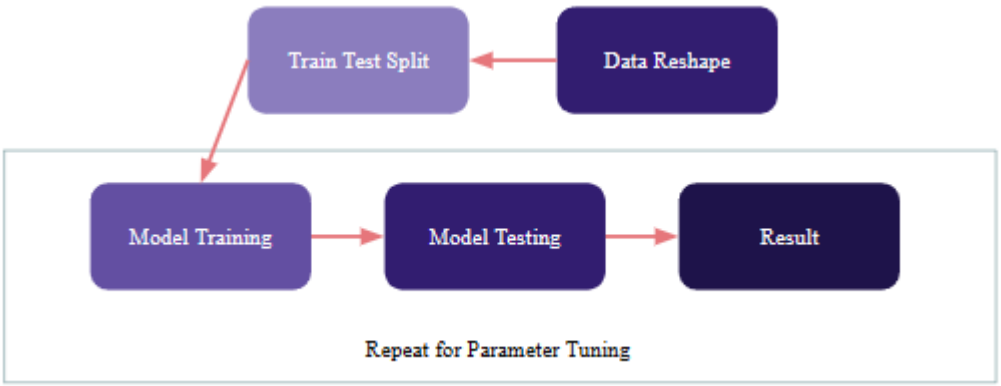


Figure 3.5 Modelling Process for CNN Model

For the CNN model, data reshape is conducted to convert the data into an array in order to be able to fit it into the CNN model. The following process is then similar to the other models.

Hybird CNN-SVM Model

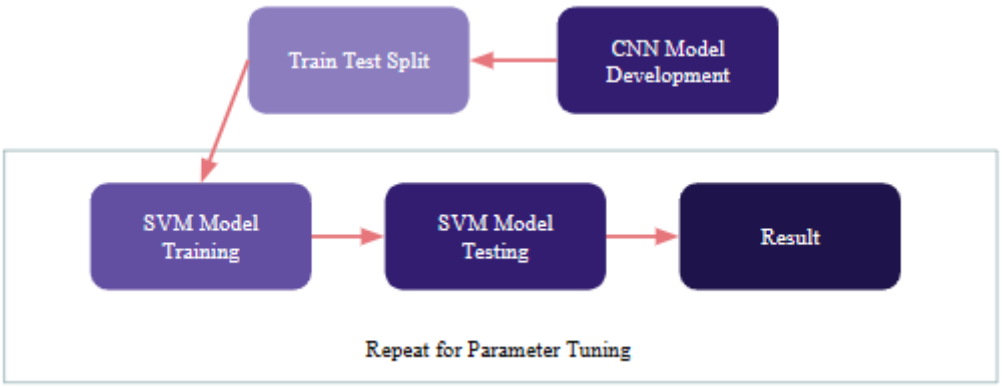


Figure 3.6 Modelling Process for Hybrid CNN-SVM Model

For the hybrid CNN-SVM model, the optimal CNN model is developed first and the model will be saved. The data after the flattening process from the CNN model is then retrieved to conduct the train test split and the following process is then similar to the normal SVM modelling process.

3.1.4 Model Performance Evaluation

The performances of the models are evaluated using the classification report and confusion matrix. By utilising the confusion matrix, the correct and incorrect predictions of the customer churn can be shown. True Positive (TP) is when the predicted result is "churn" and the actual result is also "churn", while True Negative (TN) is when the predicted result is "not churn" and the actual result is also "not churn". On the other hand, False Positive (FP) is when the model predicted the result as "churn" but the actual result is "not churn", while False Negative (FN) is when the model predicts the result as "not churn" but the actual result is "churn". From these values, the accuracy, precision value, recall value, and the F1-score can be calculated and used to know the performance of the model in predicting both the churner and non-churner. The equations below show the formula in calculating the accuracy (Eq 1), precision value (Eq 2), recall value (Eq 3), and F1-score (Eq 4):

Accuracy: $Acc = \frac{TP+TN}{TP+FP+TN+FN}$ – Eq (1)

Precision: $Prec = \frac{TP}{TP+FP}$ – Eq (2)

Recall: $Rec = \frac{TP}{TP+FN}$ – Eq (3)

F1-Score: $F1 = 2 \times \frac{Prec \times Rec}{Prec + Rec}$ – Eq (4)

3.2 Summary of Methodology & Requirement Analysis

The general framework for the churn analysis consists of 4 major processes, which starts with acquiring the dataset from the Kaggle website and analysing the dataset. Data cleaning process such as outlier detection and null value detection is done during data pre-processing and label encoding is also done to convert numerical data into textual data. MinMax Scaling is also performed as a part of data pre-processing to ensure that the model is not biased to data with larger values. Data sampling is conducted to deal with the imbalance data problem. 3 types of feature selection methods, which are Principal Component Analysis, Correlation, and Random Forest Feature Importance are also performed to identify the feature extraction method that provides the best performance. During modelling, 4 models are developed and model performance evaluation is conducted through classification report and confusion matrix.

Chapter 4: Theoretical Background

Chapter 4 Theoretical Background

This chapter discusses the methodology and theoretical background of the Multilayer Perceptron Neural Network (MLP-NN), Convolutional Neural Network (CNN), the Support Vector Machine (SVM), and the hybrid CNN-SVM model. It discusses how each algorithm functions and the elements and parameters in it.

4.1 Multilayer Perceptron Neural Network (MLP-NN)

In order to better understand a MLP Neural Network, a brief explanation towards the perceptron model is provided:

4.1.1 Single-layer perceptron model

Single-layer perceptron model is the simplest type of artificial neural networks. It is only capable of classifying linearly separable data with the output being (1 , 0) (Tyagi, 2020). It is a feed-forward type of neural network which utilizes activation functions to deliver the output. Figure 4.1 shows the architecture of a single-layer perceptron model.

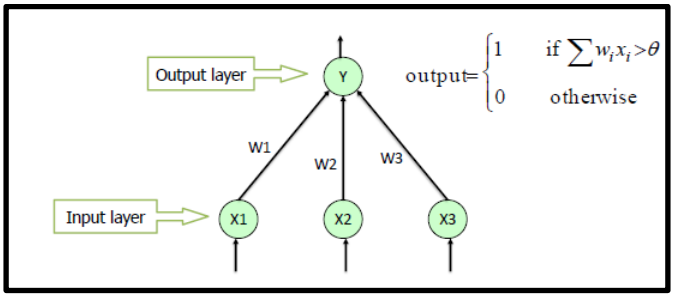


Figure 4.1 Single-layer Perceptron model (Sayad, n.d.)

In order to predict an output, weights are allocated to each of the inputs, and the algorithm will sum up all of the weighted input to generate the prediction outcome. After all of the weighted inputs, if the predicted output is above the threshold value set, then it can be assumed that the model is activated and the output value will be 1. Equation 4.1 shows how the single-layer perceptron model is able to predict the output in a binary classification case:

$$\sum w_i x_i = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n$$

$$\text{If } \sum w_i x_i > \theta \text{ Then Output = 1, otherwise output = 0}$$

Equation (4.1)

After the predicted output is obtained from the model, it will then be compared with the desired output. If the predicted output is identical to the desired output, then it can be considered that the performance of the model is satisfactory and no additional adjustments are necessary. However, if the predicted output does not match with the desired output, then the weights of

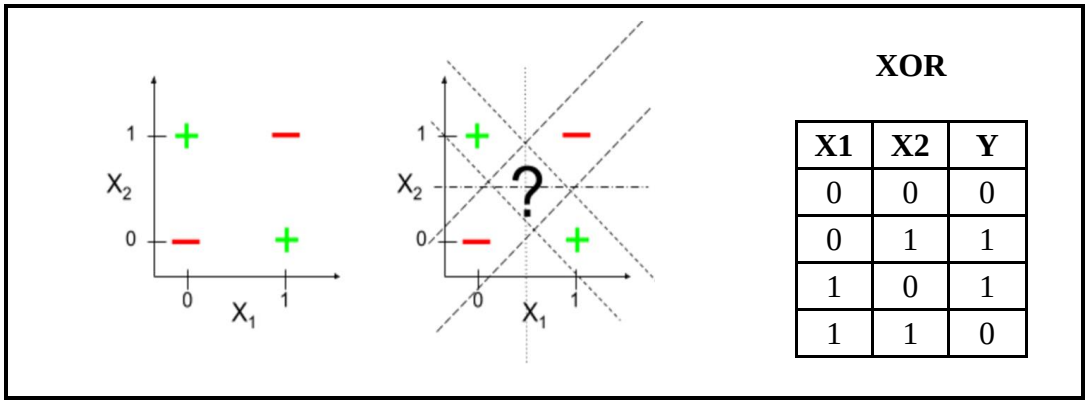
the model will need to be adjusted to minimize the errors (Sayad, n.d.). Equation 4.2 shows how the perceptron weights are adjusted to minimize the errors.

$$\Delta w = n \times d \times x$$
$$d \rightarrow Predicted\ output - Desired\ Output$$
$$\eta \rightarrow Learning\ Rate$$
$$x \rightarrow Input\ Data$$

Equation (4.2)

4.1.2 Problems with Single-layer Perceptron Model

As stated above, a single-layer perceptron model is only capable of classifying linearly separable data. Essentially, linearly separable data is the data that can be classified using a point in one dimensional model, a line in 2 dimensional model, a plane in 3 dimensional model, and so forth. However, most of the time the input data that was fed into the model will not be linearly separable, making the single-layer perceptron model not being able to suit real world cases. One of the popular examples to show this particular limitation of the single-layer perceptron model is the XOR problem (Figure 4.2).



XOR

X1	X2	Y
0	0	0
0	1	1
1	0	1
1	1	0

Figure 4.2 XOR Problem in Single-layer Perceptron Model (Battini, 2019)

Figure 4.2 illustrates a chart that represents the 2D XOR function. Since the single-layer perceptron model is only capable of classifying linearly separable data, it will not be able to classify the XOR case above successfully since the data above cannot be separated with just a single linear decision boundary (Karajgi, 2020). However, a MLP model with a backpropagation algorithm will be able to solve this problem.

4.1.3 Multilayer Perceptron Model

MLP Neural Network is a type of fully-connected neural network that contains units called network layers. Each layer in the network will consist of nodes or neurons connecting to the subsequent layers. In order for a MLP model to be trained, a minimum of 3 layers which contains the input layer, one or more hidden layer(s), and the output layer must be present in the network (Walter H. Delashmit and Michael T. Manry, 2005). Basically, it has the same structure with the single-layer perceptron model with additional hidden layers using nonlinear activation functions to adjust the weights in each iteration, thus ensuring its capability in solving non-linear separable problems. It is a type of layered feedforward neural network in which the information gained from the input will flow through each layer of the network starting from the input layer to the hidden layers, and finally the output layer unidirectionally (Taud and Mas, 2018). Figure 4.3 shows a diagram of a basic MLP Neural Network architecture.

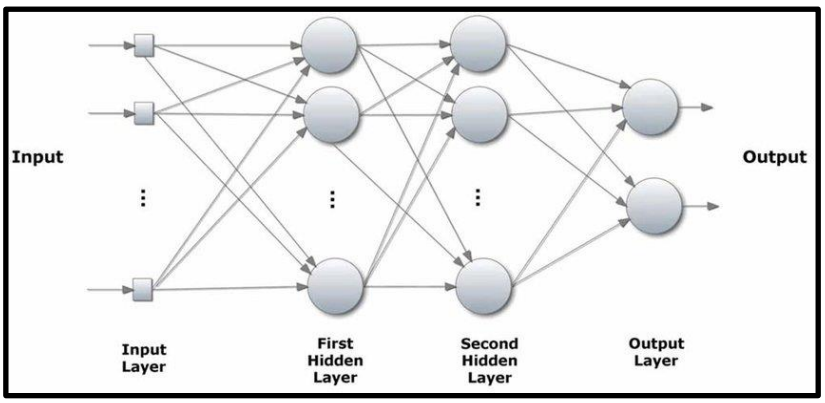


Figure 4.3: Architecture of MLP Neural Network Model with 2 Hidden Layers (Foroozesh *et al.*, 2013)

In a MLP model, the general flow to perform prediction is the same as the single-layer perceptron model, with the inclusion of hidden layer(s) between the input and output layer for the backpropagation stages. Besides, a bias(*b*), will also be added during the sum of weighted input, in order to give more degree of freedom to the MLP model. The hidden layer will utilize various different kinds of nonlinear activation function to calculate the weighted input before passing it to the subsequent layer. Linear activation function will not be utilized in the hidden layer or else the predicted output will still end up being a linearly separable solution (mlnotebook, 2017). Table 4.1 shows the different kinds of activation functions used in the hidden layers.

Table 4.1: List of Activation Functions

Activation Function	Mathematical Formula
Sigmoid	$f(x_i) = \frac{1}{1+e^{-x_i}}, f'(x_i) = \sigma(x_i)(1 - \sigma(x_i))$
Hyperbolic Tangent	$f(x_i) = tanh(x_i), f'(x_i) = 1 - tanh(x_i)^2$
Gaussian	$f(x_i) = e^{-x_i^2}, f'(x_i) = -2xe^{-x_i^2}$
Rectified Linear Unit (ReLU)	$f(x) = max(0, x)$

4.1.4 Backpropagation Algorithm

The MLP model uses a backpropagation algorithm to adjust the weights of each neuron in each iteration. The weight of each neuron is readjusted based on the calculation from the gradient of the loss function. It will update the weights starting from the last neuron layer, and work its way back to the input layer of the neural network, thus decreasing the error of the neural network. This step is repeated until the error value of the neural network is lower than the threshold value, or that the maximum iteration for the neural network is reached (Leite, 2018).

4.1.5 Tuning of MLP Model

In order to ensure that the MLP neural network is well trained, various parameters need to be fine-tuned besides the choice of variables from the training dataset to increase the performance of the neural network. The list below shows the parameters that can be tuned in a MLP model to enhance its performance:

- Number of hidden layers
- Number of nodes
- Learning Rate
- Number of iterations

A MLP model with lesser hidden layers will be able to detect nonlinear functions while having lower accuracies, however a MLP model with many hidden layers tends to overfit (Taud and Mas, 2018). The learning rate determines how much the weights of the network are adjusted based on the loss gradient. The lower the learning rate value, the slower the gradient descent (Zulkifli, 2018).

4.2 Convolutional Neural Network (CNN)

In the field of deep learning, CNN is one of the most notable and representable algorithms. It has been commonly utilized in various different kinds of fields such as image processing, computer vision, and speech recognition. One of the major advantages of employing CNN in comparison with other machine learning algorithms is that CNN is able to perform feature selection automatically without any manual user supervision. Similar to other neural networks, the structure of a common CNN algorithm is influenced by human or animal brains (Alzubaidi *et al.*, 2021). Figure 4.4 shows the schematic diagram of a basic CNN architecture.

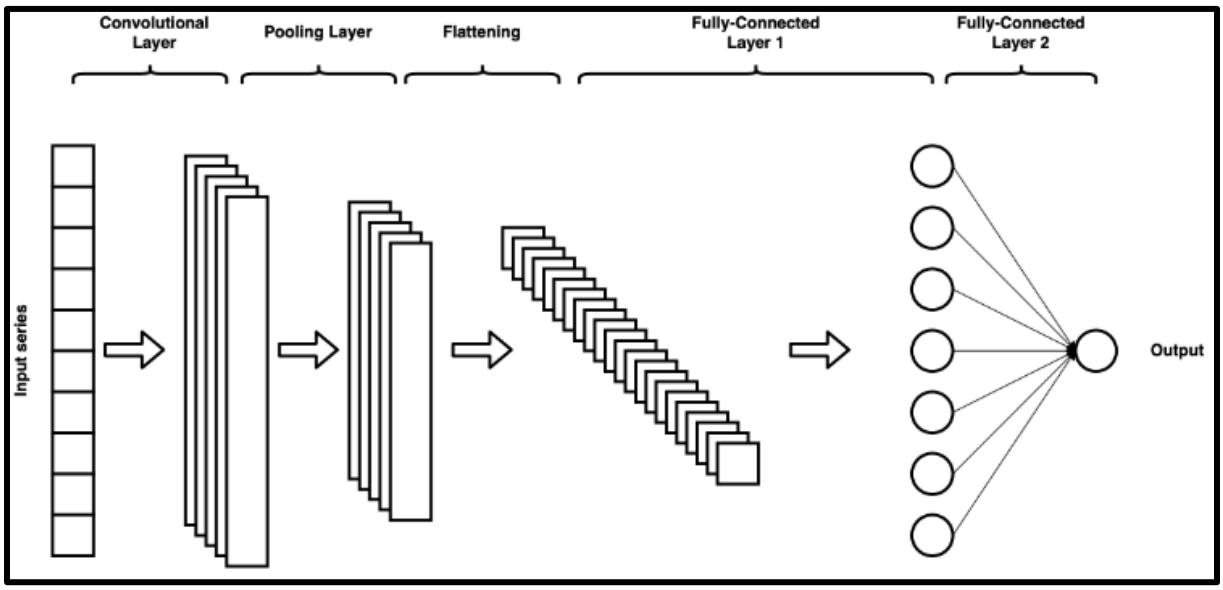


Figure 4.4 Schematic Diagram of a 1D-CNN Architecture (packt, n.d.)

Generally, the architecture of a CNN model will be categorized into 3 layers, which are the convolutional layer, pooling layer, and lastly, the fully-connected layer. The convolution layer and pooling layer will be used for the automated feature extraction, while the fully-connected layer will be used for the classification task. The inputs and neurons of each layer in the CNN models are organized into two dimensions, which are the height, and depth, in which the height will have the same value (O'Shea and Nash, 2015).

In order provide a clearer understanding about the CNN architecture, the following sections will present the detailed breakdown of each layers of the CNN model:

4.2.1 Convolutional Layer

The convolutional layer plays a major role in how CNN works. It consists of the convolutional filters, or also known as the kernels, and it is used to convolve with the input to generate the

output feature map (O'Shea and Nash, 2015). Figure 4.5 shows an example of how the kernel is employed to generate the output from the input.

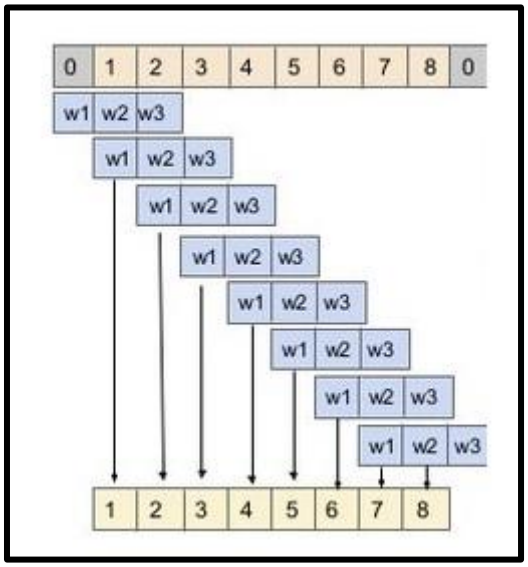


Figure 4.5: A visual representation of how the convolutional layer calculate the weighted sum of the input by employing the kernel in 1D-CNN (Bansal, 2018)

Based on Figure 4.5, it can be seen that the kernel will be multiplied with the input with the same size to the kernel to generate the output. The kernel was multiplied with the corresponding value of the input data and summed up to generate a single value. Each of these values represents an entry value to the output feature map. The same process will be performed repeatedly with the kernel moving along the input, until no further movement is possible (Alzubaidi *et al.*, 2021). Besides from using the kernel to extract the features from the input, the convolution layer is also capable of reducing the model complexity through optimizing the output. This can be done through the tuning the depth, the stride, and by setting zero-padding.

The depth is the number of kernel filters used to learn the features of each input. Reducing the depth of the output volume will significantly reduce the number of neurons in the CNN model at the same time, which enables the model to make faster computation. However, the pattern recognition capabilities of the model will also significantly deteriorate if too much of the depth is reduced.

The stride is the amount of movement that will be made by the kernel filter at a time while going through the inputs. Different strides numbers will result in different sizes of the output feature map. Based on Figure 4.5, the stride operation on input matrix with a size of 10 with

stride equals 1 is performed. As shown in the figure, if the stride is 1, the kernel filter will move along the input matrix by 1 unit from its previous position. However, it will move along by 2 units if the stride is 2. Both of these operations result in different output feature maps. Hence, smaller stride will result in frequent overlapping of the receptive fields producing outputs with larger spatial dimensions, while larger strides reduces the overlapping amount, resulting in smaller spatial dimensions of the output.

Zero-padding is the operation of adding paddings to the border of the input. This is to be able to better control the size of the spatial dimension of the output feature map (O'Shea and Nash, 2015). Figure 4.5 also shows example of zero-padding in which zero padding is added to both end of the input.

4.2.2 Activation Function

Similar to traditional neural networks, CNN also utilizes various activation functions to determine whether the information received from the neuron is important or not. It is placed between the convolutional and pooling layer in the model. It will determine whether the information is relevant, and it will ignore all of the irrelevant information (Liew, 2017). Similar to MLP, CNN model is also capable of performing nonlinear transformation through utilizing nonlinear activation functions.

4.2.3 Pooling Layer

Sub-sampling of the feature maps is conducted in the pooling layer. It aims to further reduce the dimensionality of the feature while still maintaining the major details of the features. There are various pooling techniques available to be implemented in the pooling layer, in which the most frequently utilized pooling technique is the max pooling. Other types of pooling include min pooling, and average pooling. Figure 4.6 shows an example of max pooling on a list.

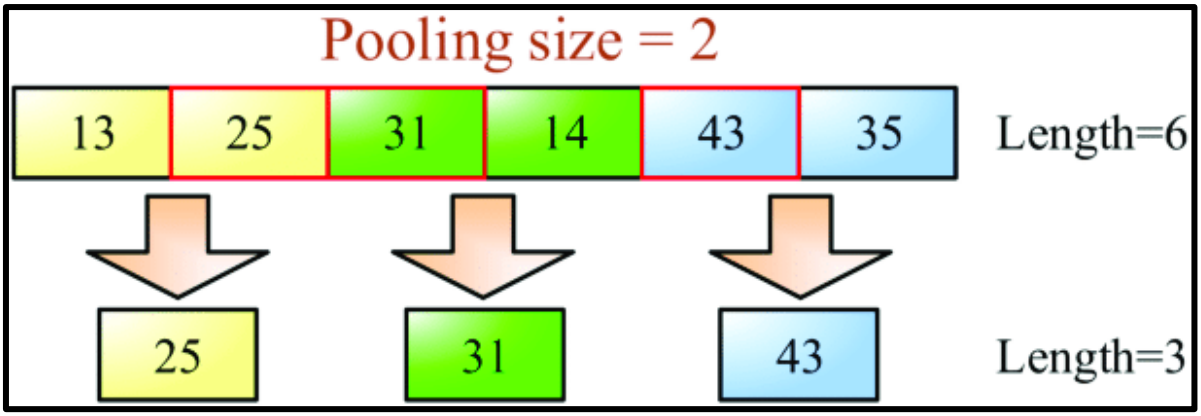


Figure 4.6 Max Pooling on a 2 matrix (ResearchGate, 2018)

Figure 4.6 shows that for the max pooling, the maximum value of the filter of size 2 is taken for the whole input matrix, thus down-sampling the length of the input from 6 into length of 3. On the other hand, another method to conduct pooling is by average pooling. In this case, the average value of the filter is taken in the operation for average pooling. Through pooling operations, the risk of overfitting the CNN model can be reduced as the number of parameters information lost will reduce the computational overhead for subsequent layers (Liew, 2017).

4.2.4 Fully-Connected Layer & Lost Functions

This layer is generally located at the end of the CNN architecture. Similar to MLP neural networks, each neuron in this layer is connected with all of the neurons in the previous layer and the subsequent layer, thus it is called the fully-connected layer (Alzubaidi *et al.*, 2021). This layer enables the CNN model to perform the classification operations after the feature maps pooled from the pooling layer are flattened. Figure 4.7 shows an example of the flattening process.

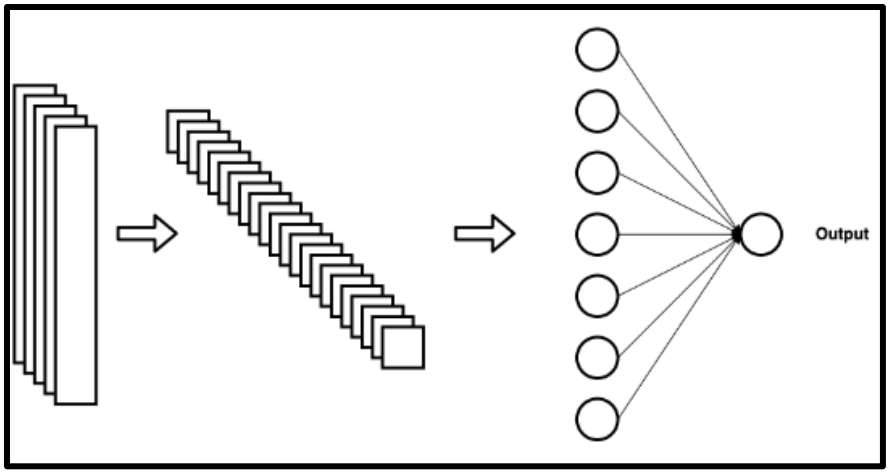


Figure 4.7 Flattening Process (packt, n.d.)

The architecture of the fully connected-layer resembles a MLP neural network (Figure 4.3), in which the input is from the previous pooling or convolutional layer. The number of possible outcomes after the classification operation depends on the number of neurons in the output layer (Liew, 2017). Similar to other neural networks, in the output layer, some lost functions may also be implemented to calculate the error between the predicted output and actual output.

4.3 Support Vector Machine (SVM)

4.3.1 SVM Model Background

SVM is a type of supervised machine learning model that is commonly used for binary classification. The main goal of a SVM model is to determine the optimal hyperplane in an N-dimensional space that is able to classify most of the number of data points correctly (Gandhi, 2018).

A hyperplane, or also called as the decision boundary, helps the SVM model to classify the data given. In a 2-dimensional space, a hyperplane is a line that separate the classes of the target variable (Stecanella, 2017). Figure 4.8 shows an example of a hyperplane separating the classes of the target variable.

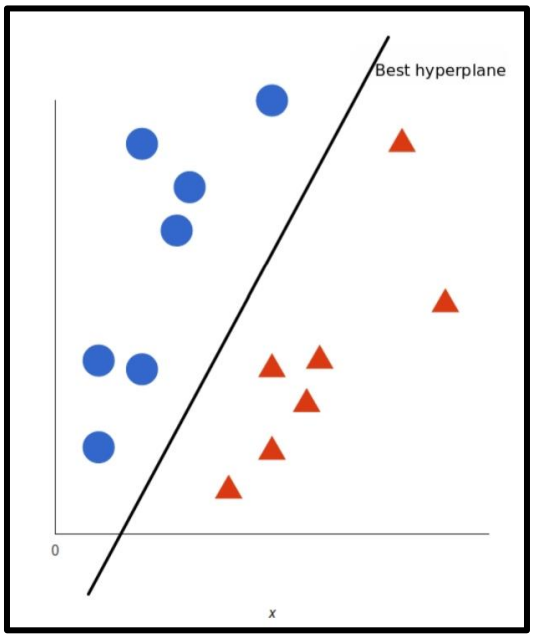


Figure 4.8 Hyperplane separating classes of target variable (Stecanella, 2017)

Based on figure 4.8, it is shown that the target variable has 2 classes, which are red and blue color. Hence, the hyperplane in this case will be a straight line separating the colors. Anything

that falls on the left side of the hyperplane will be classified as blue color, otherwise it will be classified as red color.

There are many possible hyperplanes that can be chosen. The objective of the SVM model to select the optimal hyperplane is by selecting the hyperplane that have the largest margin from both classes. By maximizing the margin, this ensures that future data points can be classified more accurately (Gandhi, 2018). Figure 4.9 shows the difference between a hyperplane with small margin and large margin.

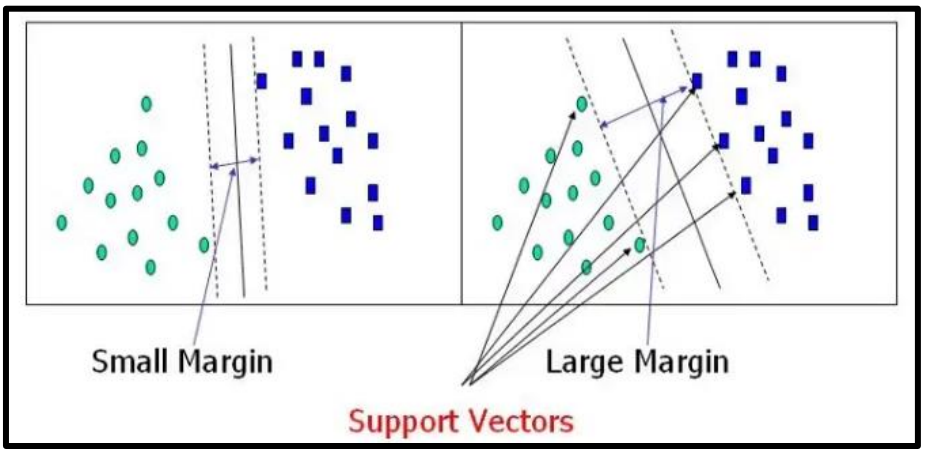


Figure 4.9 Difference between Hyperplane of Small and Large Margin (Gandhi, 2018)

Based on Figure 4.9, it is shown that the hyperplane on the right have a wider margin compared to the one on the left. This enables the model to classify the classes more accurately when new data is fed into the model as there are more space to place the data when the margin is wider.

The hyperplane of the SVM model is determined by the support vectors. Support vectors are the data points that are the nearest to the hyperplane and it will affect the position of the hyperplane (Gandhi, 2018). Figure 4.9 also shown the support vectors for both the hyperplane with small and large margins.

4.3.2 SVM Kernel

In the case when the data is not linearly separable, kernel trick may be applied to map the data to a higher dimension. Then, it will be easier to separate the data as it could be linearly divided by the hyperplane in a higher dimension. Figure 4.10 shows the mapping of a non-linearly separable data into a higher dimension.

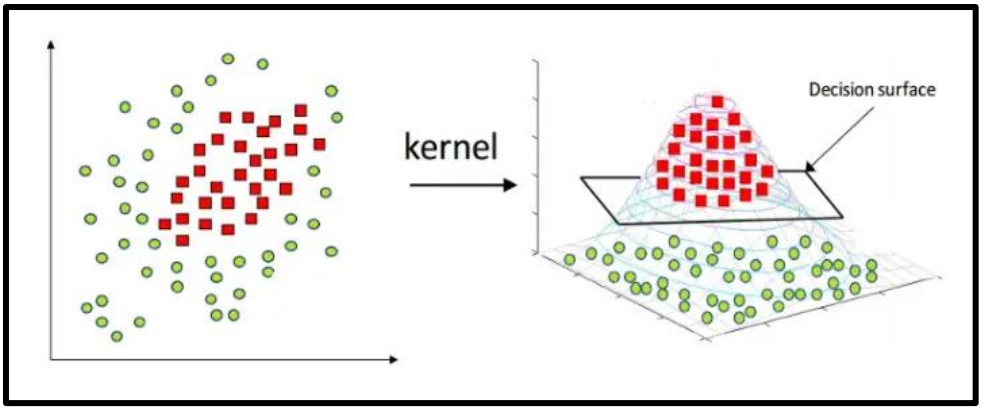


Figure 4.10 Mapping of Data into Higher Dimension (Zhang, 2018)

Based on figure 4.10, it is shown that the hyperplane can't be chosen before the data is mapped because the data is not separable. However, after the data is mapped to a higher dimension, the classes of the target variable can be separated and a hyperplane can be formed in the dimensional space. Some of the kernels most commonly used by SVM models include RBF, polynomial, linear, and sigmoid kernel.

4.4 CNN-SVM Hybrid Model

In this section, the methodology of CNN-SVM model is described in detail. There have been no applications of the CNN-SVM model in churn prediction before. This hybrid model incorporates the best features of both the CNN and SVM algorithms. As stated previously, the CNN algorithm is capable of extracting the features of the dataset in the convolutional layer and pooling layer. Hence, in the hybrid model, CNN will act as an automated feature extractor to extract the important features of the dataset. On the other hand, the SVM algorithm separates data elements belonging to different classes using a hyperplane. It is good for binary classification, however it has a poor performance when dealing with noisy data. Hence, SVM has difficulties in dealing with deep features. In the hybrid model, the SVM will act as the binary classifier to classify the results into churner and non-churner. Figure 4.11 shows an example of the architecture of a CNN-SVM hybrid model.

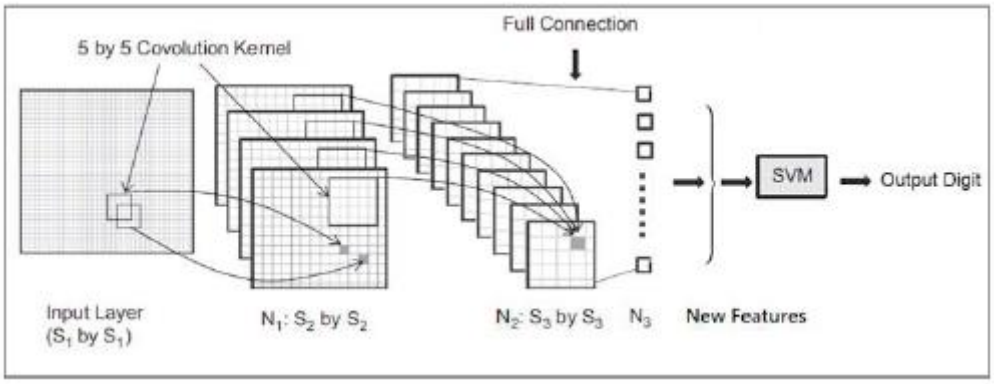


Figure 4.11 Example of Architecture of CNN-SVM hybrid architecture (Ahlawat and Choudhary, 2020)

The model will conduct a normal CNN model process first, which is feature extraction in the convolutional and pooling layer. After the fully-connected layer, the output layer will then be replaced by the SVM classifier. The features obtained from the fully-connected layer will then serve as the input for the SVM classifier.

4.5 Summary of Theoretical Background

In conclusion, a MLP neural network is a type of neural network that is capable of performing nonlinear classifications using nonlinear activation functions, thus solving the major issue faced by the single-layer perceptron model. It is categorized into three layers, which is the input layer, hidden layer(s), and output layer. It also uses a backpropagation algorithm to adjust the weight of each neuron in the network in each iteration, thus reducing the error value of the neural network.

On the other hand, a CNN is also categorized into 3 layers, which are the convolutional layer, pooling layer, and fully-connected layer. The convolutional layer and pooling layer are used for the automated feature extraction, while the fully-connected layer, which resembles a MLP neural network, is used for the classification operations. Each of the layers have hyperparameters that can be tuned to further optimize the CNN model, thus enhancing the performance of the model.

For the SVM model, it is a supervised machine learning model that aims to separate classes through a hyperplane. When the data is not linearly separable, kernel trick can be used to map the data into a higher dimension to separate the classes. Lastly, the CNN-SVM hybrid model uses CNN as the feature extractor and SVM as the classifier.

Chapter 5:
Experimental Results

Chapter 5 Experimental Results

This chapter discusses the process taken to develop the churn prediction models as well as the evaluation of the experimental results. Section 5.1 describes the data understanding process conducted on the dataset. Section 5.2 describe about the pre-processing steps taken to prepare the dataset to be fit into the models. Section 5.3 describes the experimental results of the feature extraction method, sampling technique, and the model performances. All experiments on building and testing of the models were run on an 8-core CPU AMD Ryzen 9 5900HS 3.30 GHz using Python 3. Jupyter Notebook was used to develop the models.

5.1 Data Understanding

In order to determine the factors that have major effect on whether the customer will churn or not, correlation analysis had been conducted to find out which feature correlates most with the churn variable. Since there are 20 features in the dataset chosen, only the data analysis for the top 4 features with the highest correlation were conducted. Figure 5.1 shows the heatmap that shows the top 4 features that have the highest correlation with the churn variable.

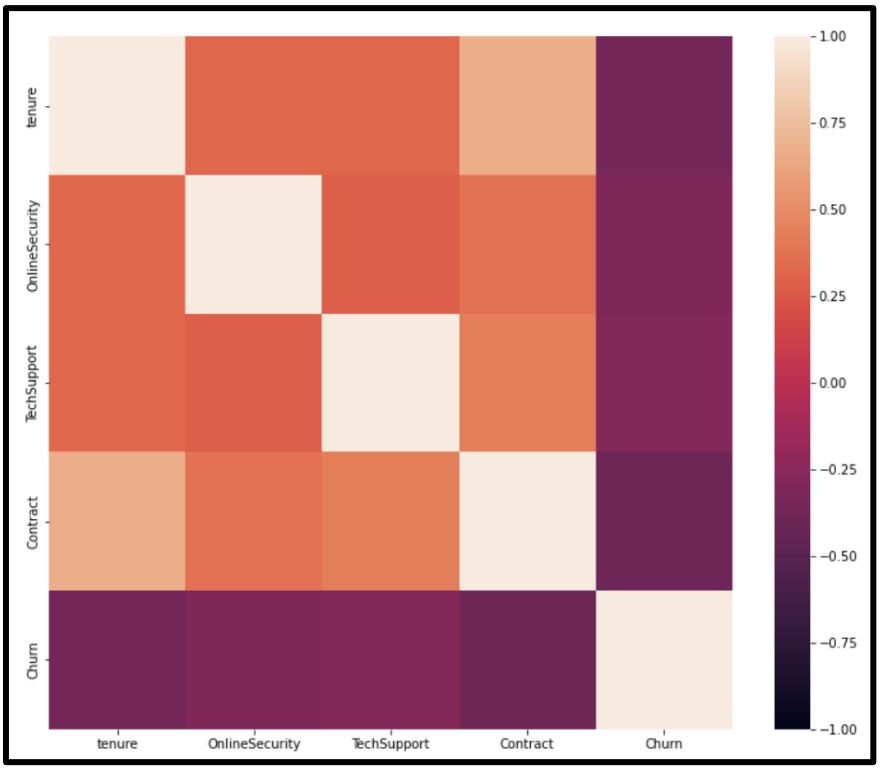


Figure 5.1 Heatmap showing Top 4 Features with Highest Correlation with Churn Variable

Based on Figure 5.1, it is shown that the feature “tenure”, “OnlineSecurity”, “TechSupport”, and “Contract” is the top 4 features that are mostly correlated with the churn variable. All these 4 features are negatively correlated to the churn variables with “Contract” having the highest negative correlation of 0.4. Figure 5.2 shows the histogram for the “tenure” count, while figure 5.3, figure 5.4, and figure 5.5 shows the count plot for the feature “OnlineSecurity”, “TechSupport”, and “Contract” respectively.

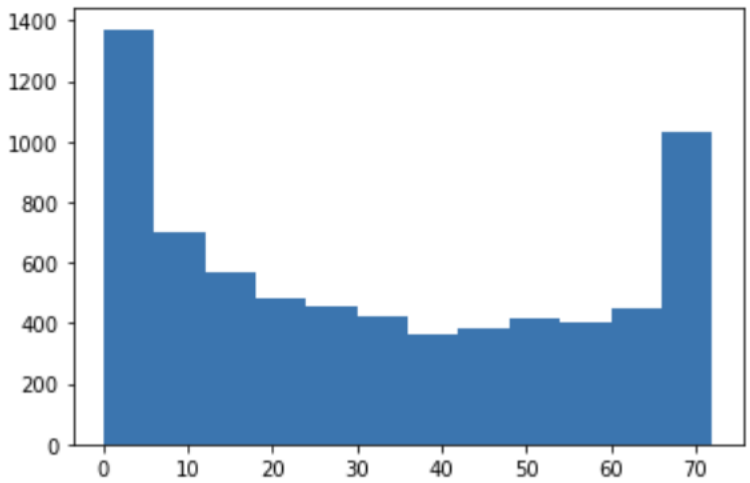


Figure 5.2 Histogram for Tenure Count

Tenure is the number of months the customer had stayed with the company. Based on Figure 5.2, it is shown that most of the customers have not stayed with the company for more than 10 months. The second highest group of customers is the customer that had stayed with the company for more than 65 months. In average, most of the customer had stayed with the company for between 20 – 60 months.

```
No:  3498
Yes:  2019
No internet Services:  1526
```

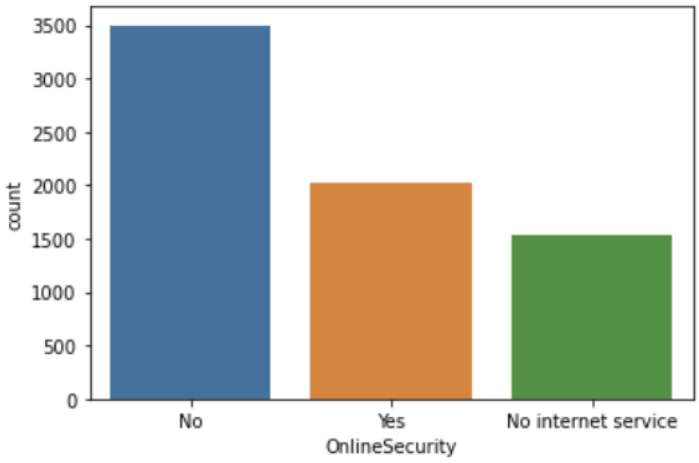


Figure 5.3 Count plot for OnlineSecurity Feature

The “OnlineSecurity” feature describes whether the customer had received online security or not from the company. Based on Figure 5.3, it is shown that most customer did not receive online security from the company with a total count of 3498 customer. On the other hand, the customer group that has the lowest count is the customer that has no internet services with a total count of 1526.

```
No:  3473
Yes:  2044
No Internet Service:  1526
```

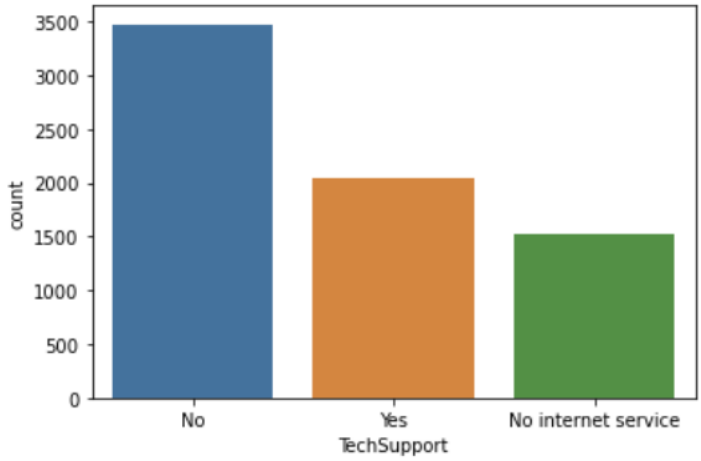


Figure 5.4 Count plot for TechSupport Feature

The “TechSupport” feature describes whether the customer had received tech support from the company. Based on Figure 5.4, it is shown that most customer did not receive tech support from the company with a total count of 3473 customer, whereas the customer group with the lowest count is the customer with no internet services, which have a total count of 1526.

```
Month-to-month:  3875
One Year:  1473
Two Year:  1695
```

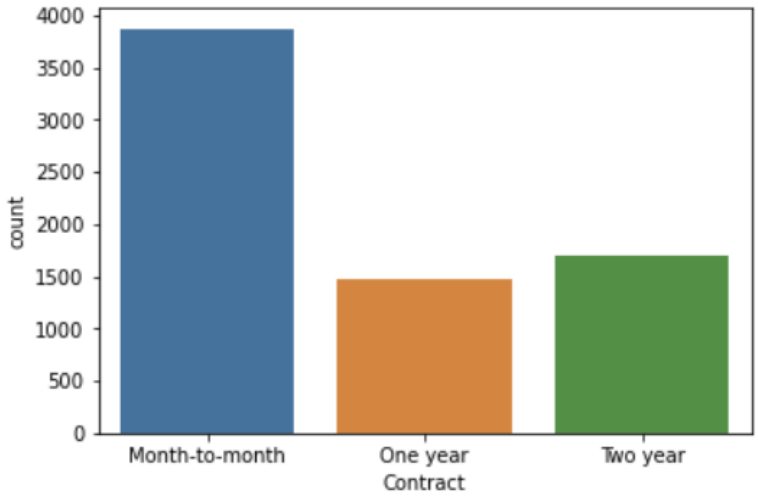


Figure 5.5 Count plot for Contract Feature

The "Contract" feature describes the type of contract the customer is having towards the company. There are three types of contract available, which are "Month-to-month", "One year", and "Two year" contracts. Based on figure 5.5, it is shown that the customer that is having a month-to-month basis contract have the highest count of 3875. On the other hand, it is shown that the customer that has a one year contract basis has the lowest count of 1473.

5.2 Data Pre-processing

After the raw data has been obtained, data pre-processing is performed to ensure that the data is suitable to perform the analysis. Firstly, data that are not important or relevant to perform churn analysis are removed. The column "CustomerID" which is only used to identify the customer is removed entirely. Processes to detect outliers was also conducted before training the model to ensure that the model trained will not be affected by it. However, there are no outliers detected in the dataset. The null value of the dataset was also identified. After converting the column "TotalCharges" from object type to float64 type, it is shown that the column has 11 null values, which is not a major occurrence. Hence, the null values were filled by the mean value of the respective column.

As the dataset selected contains a lot of textual data, primarily "yes" or "no" values in most of the columns (Figure 5.6), conversion of those data to numerical value were conducted. This is because machine learning or especially deep learning models do not work well with textual data. The Label Encoding technique was used in this case to convert the textual data into values in between 0 and n_classes-1 (Agrawal, 2018). In order to normalize the data, the MinMaxScaler technique were used to convert the numerical value of certain columns that has larger value compare to others to ensure that the model trained is not bias.

gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService
Female	0	Yes	No	1	No	No phone service	DSL
Male	0	No	No	34	Yes	No	DSL
Male	0	No	No	2	Yes	No	DSL
Male	0	No	No	45	No	No phone service	DSL
Female	0	No	No	2	Yes	No	Fiber optic
...	...	...	...	...	...	...	...
Male	0	Yes	Yes	24	Yes	Yes	DSL
Female	0	Yes	Yes	72	Yes	Yes	Fiber optic
Female	0	Yes	Yes	11	No	No phone service	DSL

Figure 5.6 Example showing dataset having a lot of textual data

In order to handle the class imbalance problem, sampling techniques were utilized. SMOTE oversampling technique and SMOTE-ENN hybrid technique were utilized to make the dataset more balanced. Performance comparison were made for three different version of the datasets, which were the original imbalanced dataset, dataset with SMOTE oversampling technique, and dataset with SMOTE-ENN hybrid technique. Table 5.1 shows the dataset versions, showing the total data, and the count of class of the churn variable.

Table 5.1 Dataset Versioning

Dataset Version	Total Data	Churn = "Yes"	Churn = "No"
Original (Imbalance)	7,043	1,869	5,174
SMOTE	10,348	5,174	5,174
SMOTE-ENN	6,096	3,353	2,743

Feature selection were also conducted on the dataset in order to build MLP models with different numbers of features. The feature selection method that will be used include principal component analysis (PCA), feature selection through correlation, and feature selection through random forest feature importance. Comparison of the model performance using different feature selection methods were also conducted.

5.3 Results

5.3.1 Comparison of Feature Extraction Methods

Comparison of the performance result has been made for each feature extraction method by using the accuracy and cross validation score of a MLP model with the same parameter setting, which have a hidden layer size of 190 and 140 for the first and second layer, with the maximum iteration value set to 10. By using the PCA feature selection method, it is shown that the model is able to achieve the optimal result when the threshold is set to 0.6 for the cumulative explained variance, which means only 60% of the dataset information is utilised to train the model and only four features are selected from the dataset. As for feature selection through correlation, it is shown that the model is able to achieve the highest performance when the minimum threshold for the correlation value is set to be larger than 0.15 and a total of 15 features are selected to train the model. Lastly, when the threshold for the importance score in the feature selection method using importance score is set to be larger than 0.03, only can the model

achieve the optimal performance with a total of 7 features being selected. It is also shown in the end that feature extraction through correlation can achieve the best result compared to other methods (Table 5.2). The MLP model can achieve the highest accuracy and cross validation score of 80% and 0.7958 respectively through correlation feature extraction method.

Table 5.2 Comparison between different feature extraction method

Methods	Accuracy	Cross Validation
Correlation	**80%**	**0.7958**
Principal Component Analysis (PCA)	78%	0.7994
Random Forest Feature Importance	79%	0.7923

5.3.2 Comparison of Sampling Method

Comparison of the performance result of various sampling method have been conducted by using the accuracy and cross validation score. Table 5.3 shows the highest performance model of each methods:

Table 5.3 Comparison between different sampling method

Dataset Version	Accuracy	Cross Validation
Original (Imbalance Dataset)	80% (MLP, CNN, CNN-SVM)	0.7980 (SVM)
SMOTE Oversampling	81% (MLP, CNN, CNN-SVM)	0.7919 (SVM)
SMOTE-ENN Hybrid Sampling	**95% (CNN-SVM)**	**0.9231 (SVM)**

Based on table 5.3, the model that are built with the dataset that utilizes the SMOTE-ENN hybrid sampling method is able to achieve the highest performance both in terms of its accuracy and cross validation score. By utilizing this sampling method, the hybrid CNN-SVM model is able to achieve the highest accuracy of 95%, whereas the SVM model is able to achieve the highest cross validation score of 0.9231. Hence, the models that are built using the SMOTE-ENN hybrid sampling dataset are used for further analysis.

5.3.3 Model Performance Evaluation

After identifying the best feature selection and sampling method to build the model, all 4 models have been developed and tuned for the best parameter settings. For the MLP model, the model is being tuned by looping the model, with increasing the neuron with each loop. The loop will only stop when the stated model accuracy is met or when the maximum neuron set in the code is reached. In the end, the MLP model achieved it's highest accuracy of 92% when it's neuron is set to 50 and 90 for the first and second layer respectively, with a maximum iteration of 30.

For the SVM model, tuning is conducted by tuning its kernel and C value. The model had gone through a looping process, with each loop having a different C value in order to find out which C value can achieve the highest accuracy. As a result, it is shown that the SVM model can achieve the best result of 93% when the C value is set to 10 and polynomial kernel is used (Figure 5.7).

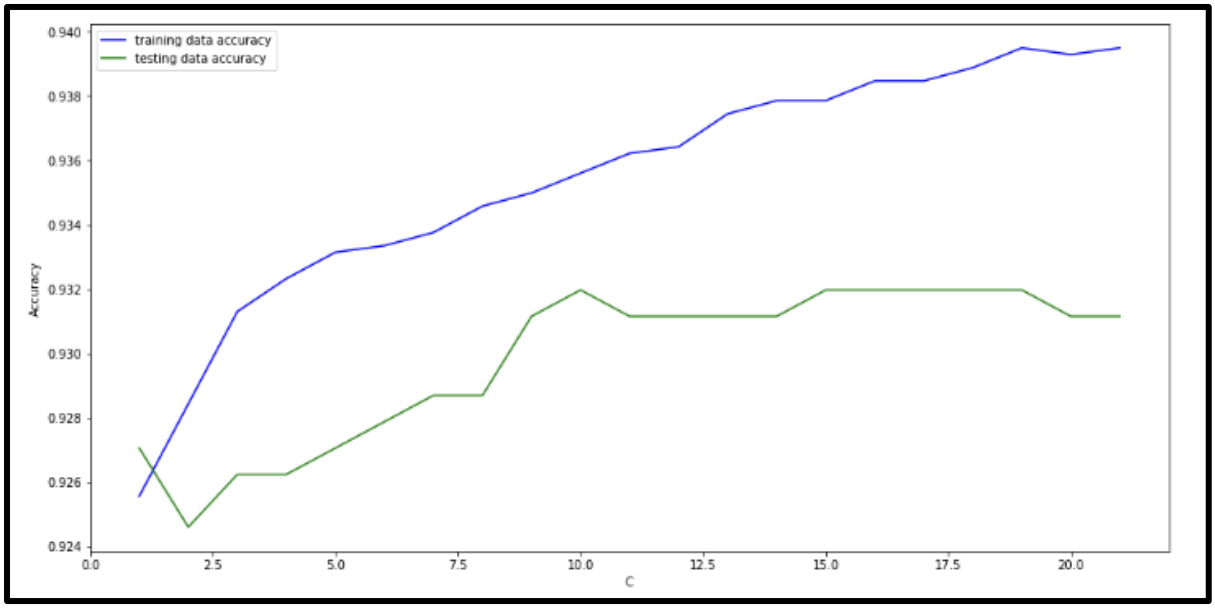


Figure 5.7 SVM model accuracy with different C value

For the CNN model, similar looping with the MLP model is also conducted to tune the neuron of the hidden layers. Besides, the epochs for the CNN model is set to 10 as the performance of the model didn't increase at further epochs, however it starts to overfit. As a result, it is shown that the CNN model is able to achieve its best model accuracy of 93% when the neurons are 780, 340, and 50 for the first, second, and third layer. Relu activation function is used in the convolutional and fully-connected layer, and sigmoid function is used for the output layer. Adam is used as the optimizer to adjust the model parameter to reduce error.

Lastly, for the hybrid CNN-SVM model, the CNN model is retrieved from the modelled built and saved previously when building the CNN model. Similar tuning process with the SVM model was then conducted for the hybrid model as it uses the SVM algorithm as a classifier. As a result, the hybrid model is able to achieve it's best model accuracy of 95% when RBF kernel is used and the C value is set to 7 (Figure 5.8).

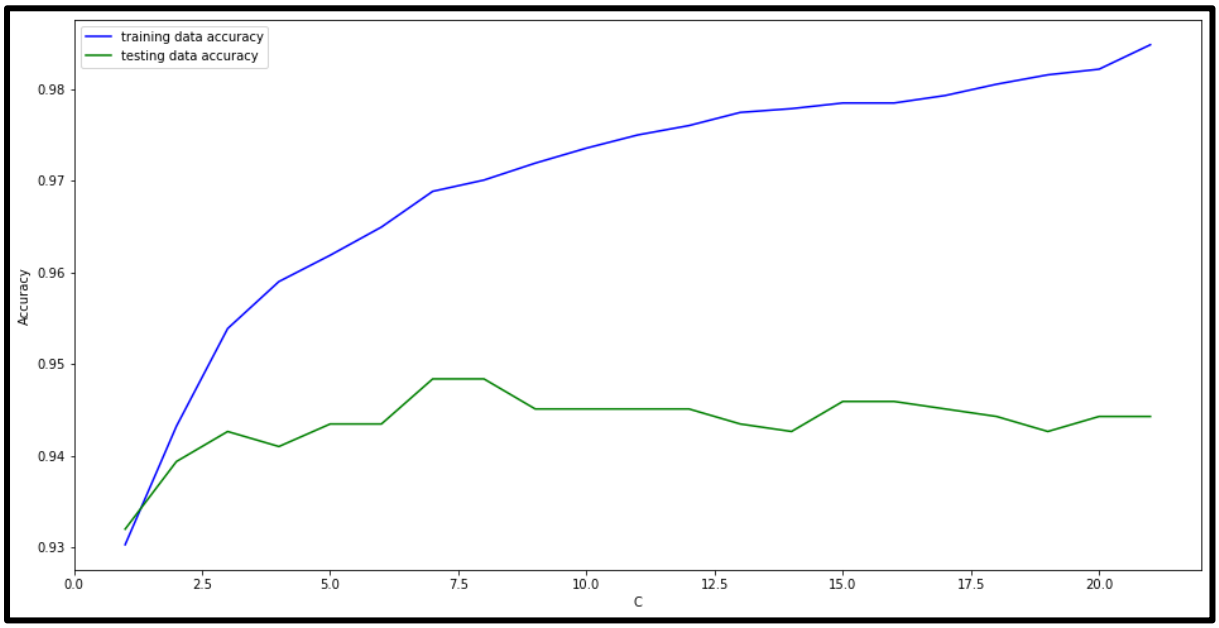


Figure 5.8 Hybrid CNN-SVM model accuracy with different C value

Comparison of the performance result of the 4 model performances built using the SMOTE-ENN hybrid sampling dataset have also been conducted. Besides from the 4 models, another 2 models from previous studies which use the same dataset are also include for comparison. Table 5.4 below shows the performance of the models:

Table 5.4 Comparison between different machine learning and deep learning models

Models	Accuracy	Cross Validation
KNN (Sjarif *et al.*, 2019)	80%	-
ANN (Agrawal, 2018)	80%	-
MLP	92%	0.9181
SVM	93%	**0.9231**
CNN	93%	-
Hybrid (CNN + SVM)	**95%**	-

Based on Table 5.4, the CNN-SVM hybrid model can achieve the highest accuracy of 95%, while the SVM model can achieve the highest cross validation score of 0.9231. The cross

validations step for the CNN and hybrid CNN-SVM model are skip to prevent excessive computational overhead.

5.3.4 Performance Evaluation

The classification report and confusion matrix showing the accuracy, precision, recall value, and F1-score of the hybrid CNN-SVM model are shown in the table 5.5 and figure 5.9 respectively below:

Table 5.5 Classification Report of Hybrid CNN-SVM Model

	Class	Precision	Recall	F1-Score	Accuracy
Hybrid Training	0	0.98	0.95	0.96	0.97
	1	0.96	0.98	0.97	
Hybrid Testing	0	0.96	0.92	0.94	0.95
	1	0.94	0.97	0.95	

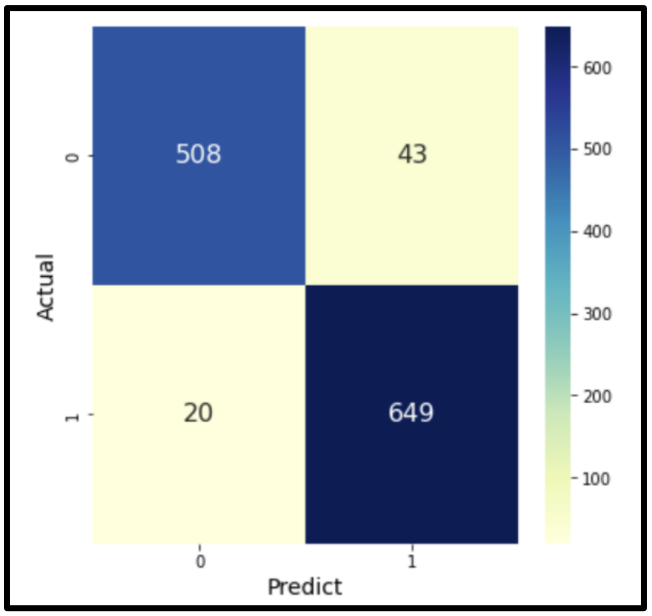


Figure 5.9 Confusion Matrix of Hybrid CNN-SVM Model

Based on the classification report on table 5.5, the recall score value of the model when predicting the customer that is not churned is the lowest. This is due to the model having the highest false negative value when predicting the customer is not churned. On the other hand, the model has the highest recall value of 0.97 when predicting the customer that have churned due to being able to classify most of the actual positive of the customer that actually churned correctly. Overall, the model is able to achieve a good performance.

Chapter 6: Discussion

Chapter 6 Discussion

In this work, the proposed methodology of comparing different machine learning and deep learning models with different feature extraction and sampling methods are completed. It is shown that the hybrid CNN-SVM model is able to achieve the highest performance compared to other models, thus proving that it is also useful in conducting churn prediction, and that the structural risk minimization technique used by SVM model can achieve higher performance when combined with the automated feature extraction feature of the CNN model.

In table 5.2, the MLP model using the correlation feature extraction method is shown to be having the highest performance compared to the PCA and random forest feature importance method. This may be due to when using the correlation method, most of the feature was fit into the MLP model when compared to the other method, causing the model to be able to retrieve more information about the dataset to train the model.

During the comparison of the sampling method used as shown in table 5.3, the SMOTE-ENN hybrid sampling method is shown to be able to achieve the best performance compared to the original dataset and the SMOTE oversampling method. The reason why the model performance only slightly increased after conducting the SMOTE oversampling technique compared to the original dataset may be because after SMOTE has been utilized, the class clusters may be invading each other spaces, therefore causing the model to not be able to have a good decision boundary separating the classes and thus causing more misclassification. Hence, after ENN undersampling method is utilized, extensive cleaning is performed on the dataset which can increase the class separation near the decision boundary, thus allowing the model to have a more clear and concise class separation and decreasing the misclassification of the target variable (Satpathy, 2021)

Lastly, in table 5.4, it is shown that the models from the previous studies both achieved an accuracy of 80% for their model. This may be due to no sampling method is applied in their studies as the performance of the model built in this study with the imbalance dataset is having similar accuracy as shown in table 5.3.

Chapter 7: Conclusion

Chapter 7 Conclusion

This study presented the development of a hybrid CNN-SVM model in conducting churn analysis. Other models which are MLP, SVM, and CNN models are also built for comparison with various feature extraction and sampling method. After training and building all of the models, it is shown that the hybrid CNN-SVM model thus indeed achieved the highest performance with a test accuracy of 95% when the SMOTE-ENN hybrid sampling method is used to solve the dataset class imbalance problem. Besides, it is also shown that the feature "tenure", "OnlineSecurity", "TechSupport", and "Contract" are the 4 major factors that affect whether a customer will churn or not.

The main contribution of this study is to develop the hybrid CNN-SVM model that is capable of conducting the churn analysis, in which this study had successfully completed. Therefore, real world company may implement the same model to have a better churn prediction model, thus being capable on focusing more on the correct potential customer churn to prevent the loss of revenue. The limitation of this study is that the model performance is only evaluated using only one dataset. This prevent us from ensuring the model performance will not deteriorate when data from different datasets are fit into the models. This is because there are various types data for a churn dataset. Some datasets might even include time series data, which the models built may not work well with.

Future works of this study consists of comparing the model performance between different datasets. As stated above, the model only works well with the dataset used in this study, but the model performance on other churn datasets have not been evaluated. By comparing the model performance with different datasets, we may ensure that whether the model is able to work well in the actual work environment instead of just in theory. Besides, cooperation with real world telecommunication company is recommended in order to get the most updated and accurate dataset currently.

References

Agrawal, S. (2018) 'Customer Churn Prediction Modelling Based on', *2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE)*, pp. 1–6.

Ahlawat, S. and Choudhary, A. (2020) 'Hybrid CNN-SVM Classifier for Handwritten Digit Recognition', *Procedia Computer Science*, 167(2019), pp. 2554–2560. doi: 10.1016/j.procs.2020.03.309.

Ahn, J. *et al.* (2020) 'A Survey on Churn Analysis in Various Business Domains', *IEEE Access*, 8, pp. 220816–220839. doi: 10.1109/ACCESS.2020.3042657.

Albawi, S., Mohammed, T. A. M. and Alzawi, S. (2017) 'Layers of a Convolutional Neural Network', *Ieee*, p. 16.

Alboukaey, N., Joukhadar, A. and Ghneim, N. (2020) 'Dynamic behavior based churn prediction in mobile telecom', *Expert Systems with Applications*, 162, p. 113779. doi: 10.1016/j.eswa.2020.113779.

Alzubaidi, L. *et al.* (2021) *Review of deep learning: concepts, CNN architectures, challenges, applications, future directions*, *Journal of Big Data*. Springer International Publishing. doi: 10.1186/s40537-021-00444-8.

Amuda, K. A. and Adeyemo, A. B. (2019) 'Customers Churn Prediction in Financial Institution Using Artificial Neural Network'. Available at: http://arxiv.org/abs/1912.11346.

Bansal, S., 2018. *kaggle.* [Online]
Available at: https://www.kaggle.com/code/shivamb/3d-convolutions-understanding-use-case
[Accessed 21 12 2022].

Battini, D., 2019. *Tech-Quantum.* [Online]
Available at: https://www.tech-quantum.com/solving-xor-problem-using-neural-network-c/
[Accessed 16 8 2022].

Bayrak, A. T. *et al.* (2021) 'Personalized customer churn analysis with long short-term memory', *Proceedings - 2021 IEEE International Conference on Big Data and Smart Computing, BigComp 2021*, pp. 79–82. doi: 10.1109/BigComp51126.2021.00024.

BRANDUSOIU, I. and TODEREAN, G. (2013) 'Churn Prediction in the Telecommunications Sector Using Support Vector Machines', *ANNALS OF THE ORADEA UNIVERSITY. Fascicle of Management and Technological Engineering.*, XXII (XII)(1). doi: 10.15660/auofmte.2013-1.2772.

Çelik, Ö. and Osmanoğlu, U. (2019) 'Comparing to Techniques Used in Customer Churn Analysis', *Journal of Multidisciplinary Developments*, 4(1), pp. 30–38. Available at: https://www.researchgate.net/publication/337103029.

Dingli, A., Marmara, V. and Fournier, N. S. (2017) 'Comparison of deep learning algorithms to predict customer churn within a local retail industry', *International Journal of Machine Learning and Computing*, 7(5), pp. 128–132. doi: 10.18178/ijmlc.2017.7.5.634.

Farquad, M. A. H., Ravi, V. and Raju, S. B. (2014) 'Churn prediction using comprehensible support vector machine: An analytical CRM application', *Applied Soft Computing Journal*, 19, pp. 31–40. doi: 10.1016/j.asoc.2014.01.031.

Foroozesh, J. *et al.* (2013) 'Application of Artificial Intelligence (AI) Modeling in Kinetics of Methane Hydrate Growth', *American Journal of Analytical Chemistry*, 04(11), pp. 616–622. doi: 10.4236/ajac.2013.411073.

Gandhi, R., 2018. *Towards Data Science.* [Online]
Available at: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47
[Accessed 14 12 2022].

Ismail, M. R. *et al.* (2015) 'A multi-layer perceptron approach for customer churn prediction', *International Journal of Multimedia and Ubiquitous Engineering*, 10(7), pp. 213–222. doi: 10.14257/ijmue.2015.10.7.22.

J.Alsakran, M. H. A. R. A. T. (2019) 'Churn Prediction: A Comparitive Study Using KNN and Decision Trees', *IEEE Access*, (1), pp. 182–186.

Jain, H., Khunteta, A. and Srivastava, S. (2020) 'Churn Prediction in Telecommunication using Logistic Regression and Logit Boost', *Procedia Computer Science*, 167(2019), pp. 101–112. doi: 10.1016/j.procs.2020.03.187.

Joshi, P. and Gupta, S. (2019) 'Predicting Customers Churn in Telecom Industry using Centroid Oversampling method and KNN classifier', *International Research Journal of Engineering and Technology*, pp. 3708–3712. Available at: www.irjet.net.

Kaggle, n.d. *Kaggle.* [Online]
Available at: https://www.kaggle.com/datasets/blastchar/telco-customer-churn
[Accessed 16 8 2022].

Karajgi, A., 2020. *towardsdatascience.* [Online]
Available at: https://towardsdatascience.com/how-neural-networks-solve-the-xor-problem-59763136bdd7
[Accessed 16 8 2022].

Karanovic, M. *et al.* (2018) 'Telecommunication Services Churn Prediction - Deep Learning Approach', *2018 26th Telecommunications Forum, TELFOR 2018 - Proceedings*, pp. 420–425. doi: 10.1109/TELFOR.2018.8612067.

Kaur, I. and Kaur, J. (2020) 'Customer churn analysis and prediction in banking industry using machine learning', *PDGC 2020 - 2020 6th International Conference on Parallel, Distributed and Grid Computing*, pp. 434–437. doi: 10.1109/PDGC50313.2020.9315761.

Leite, T. M., 2018. *medium.com.* [Online]
Available at: https://medium.com/@tiago.tmleite/neural-networks-multilayer-perceptron-and-the-backpropagation-algorithm-a5cd5b904fde
[Accessed 16 8 2022].

Liew, A. (2017) 'Gesture Recognition-Malaysian Sign Language Recognition with Convolutional Neural Network'.

Luo, B., Shao, P. and Liu, J. (2007) 'Customer churn prediction based on the decision tree in personal handyphone system service', *IEEE Access*. doi: 10.1109/ICSSSM.2007.4280145.

Mena, C. G. *et al.* (2019) 'Churn Prediction with Sequential Data and Deep Neural Networks. A Comparative Analysis', pp. 1–12. Available at: http://arxiv.org/abs/1909.11114.

Mishra, A. and Reddy, U. S. (2017) 'A Novel Approach for Churn Prediction Using Deep Learning', *2017 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2017*, pp. 1–4. doi: 10.1109/ICCIC.2017.8524551.

mlnotebook, 2017. *mlnotebook.github.io.* [Online]
Available at: https://mlnotebook.github.io/post/transfer-functions/
[Accessed 16 8 2022].

O'Shea, K. and Nash, R. (2015) 'An Introduction to Convolutional Neural Networks', pp. 1–11. Available at: http://arxiv.org/abs/1511.08458.

packt, n.d. *packt.* [Online]
Available at:
https://subscription.packtpub.com/book/data/9781789618518/10/ch10lvl1sec63/convolutional-neural-networks-for-time-series-forecasting
[Accessed 20 12 2022].

Pejić Bach, M., Pivar, J. and Jaković, B. (2021) 'Churn Management in Telecommunications: Hybrid Approach Using Cluster Analysis and Decision Trees', *Journal of Risk and Financial Management*, 14(11), p. 544. doi: 10.3390/jrfm14110544.

Peng, L. L. S. B. T. L. Y. (2014) 'Telecom Customer churn prediction based on cluster stratified sampling logistic regression', *International Journal of Digital Content Technology and its Applications*, 5(10), pp. 381–388. doi: 10.4156/jdcta.vol5.issue10.45.

ResearchGate, 2018. *ResearchGate.* [Online]
Available at: https://www.researchgate.net/figure/1D-max-pooling-operation_fig4_324177888
[Accessed 21 12 2022].

Satpathy, S., 2021. *Analytics Vidhya.* [Online]
Available at: https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/
[Accessed 21 12 2022].

Sayad, S., n.d. *saedsayad.com.* [Online]
Available at: https://www.saedsayad.com/artificial_neural_network_bkp.htm
[Accessed 16 8 2022].

Sjarif, N. N. A. *et al.* (2019) 'A customer Churn prediction using Pearson correlation function and K nearest neighbor algorithm for telecommunication industry', *International Journal of Advances in Soft Computing and its Applications*, 11(2), pp. 46–59.

Stecanella, B., 2017. *MonkeyLearn.* [Online]
Available at: https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/
[Accessed 24 12 2022].

Taud, H. and Mas, J. F. (2018) 'Multilayer Perceptron (MLP)', pp. 451–455. Available at: https://doi.org/10.1007/978-3-319-60801-3_27.

Townsend, A. and Nilakanta, S. (2019) 'Customer Churn : A Study of Factors Affecting Customer Churn using Machine Learning', *Creative Component*, pp. 1–30.

Tsai, C. F. and Lu, Y. H. (2009) 'Customer churn prediction by hybrid neural networks', *Expert Systems with Applications*, 36(10), pp. 12547–12553. doi: 10.1016/j.eswa.2009.05.032.

Tyagi, N., 2020. *medium.com.* [Online]
Available at: https://medium.com/analytics-steps/understanding-the-perceptron-model-in-a-neural-network-2b3737ed70a2#:~:text=In%20the%20Artificial%20Neural%20Network,classifiers%20in%20supervised%20machine%20learning
[Accessed 16 8 2022].

Umayaparvathi, V. and Iyakutti, K. (2017) 'Automated Feature Selection and Churn Prediction using Deep Learning Models', *International Research Journal of Engineering and Technology(IRJET)*, 4(3), pp. 1846–1854. Available at: https://irjet.net/archives/V4/i3/IRJET-V4I3422.pdf.

Walter H. Delashmit and Michael T. Manry (2005) 'Recent developments in multilayer perceptron neural networks', *Proceedings of the 7th Annual Memphis Area Engineering and Science Conference*, (MAESC). Available at: https://www.researchgate.net/publication/228651431.

Xia, G. E. and Jin, W. D. (2008) 'Model of customer churn prediction on support vector machine', *Xitong Gongcheng Lilun yu Shijian/System Engineering Theory and Practice*, 28(1), pp. 71–77. doi: 10.1016/s1874-8651(09)60003-x.

Xie, Y. *et al.* (2009) 'Customer churn prediction using improved balanced random forests', *Expert Systems with Applications*, 36(3 PART 1), pp. 5445–5449. doi: 10.1016/j.eswa.2008.06.12

Zhang, G., 2018. *Medium.* [Online]
Available at: https://medium.com/@zxr.nju/what-is-the-kernel-trick-why-is-it-important-98a98db0961d
[Accessed 24 12 2022].

Zulkifli, H., 2018. *towardsdatascience.com.* [Online]

Available at: https://towardsdatascience.com/understanding-learning-rates-and-how-it-improves-performance-in-deep-learning-d0d4059c1c10

[Accessed 16 8 2022].

Appendix

Dataset Link: https://www.kaggle.com/datasets/blastchar/telco-customer-churn