ANALYSIS OF CONVOLUTIONAL NEURAL NETWORK ON CUSTOMER CHURN PREDICTION

TAN YAN LIN

SESSION 2022/2023

FACULTY OF INFORMATION SCIENCE & TECHNOLOGY

MULTIMEDIA UNIVERSITY

JUNE 2023

ANALYSIS OF CONVOLUTIONAL NEURAL NETWORK ON CUSTOMER CHURN PREDICTION

BY

TAN YAN LIN

SESSION 2022/2023

THE PROJECT REPORT IS PREPARED FOR

FACULTY OF INFORMATION SCIENCE & TECHNOLOGY
MULTIMEDIA UNIVERSITY
IN PARTIAL FULFILLMENT
FOR

BACHELOR OF INFORMATION TECHNOLOGY
B.I.T. (HONS) SECURITY TECHNOLOGY

FACULTY OF INFORMATION SCIENCE & TECHNOLOGY

MULTIMEDIA UNIVERSITY

JUNE 2023

DECLARATION

I hereby declare that the work have been done by myself and no portion of the work contained in this thesis has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

TAN YAN LIN

Faculty of Information Science & Technology
Multimedia University

Date: 7.6.2023

ACKNOWLEDGEMENT

I would like to express my heartfelt thanks to my supervisor, Dr. Pang Ying Han, for her enthusiasm, support, patience, invaluable guidance, and endless ideas throughout my final year project. Thanks to her vast knowledge and expertise, I am able to successfully complete the project.

In addition, I would also like to thank my loving parents and friends, and all those who have helped and supported me. Without their support, I would not be able to overcome the challenge and completed this project.

Finally, I would like to express our gratitude to Multimedia University for providing a conducive learning environment and making the final year project as a compulsory course. It is a great pleasure for us to gain deeper knowledge in related fields.

ABSTRACT

In telecommunication industry, predicting customer churn is crucial for enhancing the customer retention. However, accurate churn prediction in this industry is challenging. This paper aims to analysis on convolutional neural network on churn prediction. Nevertheless, single classifiers can limit the potential for effectiveness of churn prediction. Therefore, this paper presented a stacking ensemble model for churn prediction in the telecom industry to find the optimal method.

In order to process the project, several research have been done to understand the process of churn prediction. The finding of the researchers is included in the literature review. It is crucial to critically analyse the studies to determine their relevance and methodology.

The proposed model consists of various components to ensure the effectiveness of the model. These components include data pre-processing and exploratory data analysis (EDA) to enhance data quality and gain insights from the dataset. Additionally, resampling techniques such as Synthetic Minority Oversampling Technique (SMOTE) are adopted to address imbalanced nature on datasets which is commonly occurs in churn prediction tasks.

The core component of the model is the churn prediction in where the modelling involve. Multiple trained base classifiers such as Convolutional Neural Network (CNN), Logistic Regression, Decision Tree, and Support Vector Machine (SVM) are combined into a meta learner (CatBoost) using stacking ensemble. The model is evaluated. In stacking ensemble, the model leveraging the diverse strengths of individual classifiers to the ensemble model.

The proposed model is tested and evaluated using a public dataset, Cell2cell dataset and well-known metrics. In Cell2cell dataset, the model achieved accuracy of 60.62%, precision of 66.57%, recall of 60.62%, f1-score of 62.40%, and AUC of 63%. However, it is advisable to compare the model with other existing models to ensure the effectiveness and performance of the model.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS/ SYMBOLS

FYP	Final Year Project
FIST	Faculty of Information Science & Technology
CNN	Convolutional Neural Network
CCP	Customer Churn Predicition
AUC	Area Under Curve
LR	Logistic Regression
DT	Decision Tree
SVM	Support Vector Machine
EDA	Exploratory Data Analysis
VAE	Variational Autoencoder
DL	Deep Learning
SMOTE	Synthetic Minority Oversampling Technique
Acc	Accuracy
ROC	Receive Operating Characteristic Curve

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Overview

In today's interconnected world, telecommunication has assumed even greater significance, with the widespread adoption of mobile phones contributing to an exponential surge in its importance. The telecommunication industry has become an essential part of our daily lives, as evidenced by the fact that the number of mobile phone users with cellular subscriptions in 2022 surpasses the global population, according to data provided by the International Telecommunication Union (Facts and Figures 2022 - Subscriptions, accessed May 29, 2023). Countries from the CIS region has roughly 3 cellular subscriptions for every two citizens. Thus, the telecommunication market is immensely saturated as it has to accommodate at least the world's population which is as of now 8 billion people. Companies have recognized the value of venturing into the telecommunication market, leading to the incorporation of numerous telecom service providers. If telecom service providers fail to retain their customers, they may switch to another service provider, resulting in churn.

Churn leads to potential revenue and profit loss for the company. Churn prediction is a useful tool to predict customer at churn risk (Abbasimehr et al., 2011). With churn prediction the telco service providers would be able to predict and identify the customers who are likely of churning, and they may strategize against it in order to retain the churning customers which would result in maintaining a stable profit in an unstable industry. Predicting customer churn involves analysing relevant customer data as various factors can influence the churn rate. Some commonly used single classifiers are decision trees, logistic regression, support vector machines, random forests, and neural networks.

In this study, the primary focus is to enhance the Convolutional Neural Networks (CNN) in predicting customer churn in the telecommunication industry. In order to achieve better accuracy, ensemble learning is employed by combining CNN with other machine learning classifier. This approach leverages the strengths of multiple classifiers to compensate for weakness of each classifier to enhance the overall performance. The performance of the model was evaluated and comparison is made to ensure the performance of the mode.

1.2 Problem Statement

In highly competitive industry, customer retention shows a significant important for telecommunication service providers. In order for the telecom service providers to ensure they are in the market they would have to face the issue of customer retention in order to maintain stable profit continue staying in the market. It is much more beneficial for telecom service providers to focus on retaining existing customers rather than acquiring new clients (De Bock & Poel, 2011). Hence, effective customer churn prediction is crucial. Churn prediction enables telecom service providers to identify and predict customers who are likely to churn, allowing them to strategize to retain them.

However, predicting customer churn poses challenges on analysing the relevant customer data. In churn predictions, traditional machine learning was widely used. Each traditional machine learning has its own weaknesses in capturing the complex relationship of churn data which might have impact on predicting churn. It is crucial to enhance the accuracy of customer churn prediction in the telecommunication industry. This study aims to exploring CNN which is highly effective in analysing complex data to improving churn prediction. Furthermore, ensemble learning techniques will be employed to combine the CNN with other machine learning classifiers. This approach can help compensate the weaknesses of each classifier and improve the overall churn prediction accuracy.

1.3 Project Objectives

This project consist of two objectives as follows:

- To study the Convolutional Neural Network on customer churn prediction.

- To improve the performance of churn prediction using Convolutional Neural Network.

1.4 Project Scope

The scope of this project involves in several key stages which aim to enhance churn prediction. The proposed work involves Exploratory Data Analysis, data pre-processing, class imbalance handling, model building and evaluation. The project initializes with EDA to gain insight from the dataset. Then, the pre-processing step consists of handle missing and irrelevant data and perform data transformation. Next, handling imbalance classes using SMOTE is carried out to balance the dataset. Then, the next step is the most important steps which is building the predictive model. The model was using Logistic Regression, Decision Tree, Support Vector Machine, Convolutional Neural Network, and ensemble to stacking classifier. The model is evaluated by the unseen dataset. The proposed model is evaluated on the public dataset, Cell2cell.

1.5 Project Management

Table 1.1 show the Gantt chart for FYP 1.

Table 1.1: Gantt chart for FYP 1

	1	2	3	4	5	6	7	8	9	10	11	12
Data Preparation												
Literature Review												
Data Pre-processing												
Exploratory Data Analysis												
Feature Engineering												
Resampling												
Modelling												
Evaluate Model												
Model Enhancement												
Journal Paper Writing												
Report Writing												

1.6 Report Organisation

This paper is organized in five Chapter. Chapter 1 serves as an introduction to churn prediction in telecommunication industry, followed by the problem statement, objectives, and project management addressed in this paper. In Chapter 2, comprehensive literature review is conducted by research on relevant articles that related to the proposed model for churn prediction. Chapter 3 describe the methodology adopted in the experiment, which outlining the approach taken to build a churn prediction model. Next, Chapter 4 present the experiment set up and analysis on the result obtained from the experiment. Finally, Chapter 5 presents the comparison between proposed model with the existing model, summarize the finding of the paper, and suggestion for future work suggestion.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Huge number of research have been conducted on the customer churn prediction through various of methods. This chapter aims to provide the overview of the different methods that haven been developed by exploring previous conducted studies on churn prediction. This chapter is structured into 2 parts, section 2.2 will discuss on machine learning on churn prediction., section 2.3 will present on ensemble learning on churn prediction, and section 2.4 will present on Deep Learning on churn prediction.

2.2 Customer Churn Prediction in Machine Learning

Various machine learning methods have been adopted in churn prediction. Idris et al. (2012) developed a particle swarm optimization (PSO)-based undersampling methods to address the issue of imbalance nature data distribution in churn prediction. After they evaluate different reduction techniques, they discovered that the approach integrated with Random Forest (RF) together with K-Nearest Neighbor (KNN) classifiers, produced better Area Under Curve (AUC) values. Their proposed approach based on PSO, mRMR, and RF termed as Chr-PmRF results achieved the best AUC value of 75.11%.

Wu et al. (2021) explored different machine learning classifier on multiple datasets on customer segmentation and customer churn prediction. The experiment adopted six classifiers which are Logistic Regression, Decision Tree, Random Forest (RF), Naïve Bayes, AdaBoost, Multi- Layer Perceptron and three datasets. Their experiment was conducted with and without Synthetic Minority Oversampling Technique (SMOTE) method. The best performance model is for dataset 1 without SMOTE is AdaBoost, dataset 2 with SMOTE is Random Forest, and dataset 3 with SMOTE is Random Forest in terms of accuracy.

Moreover, Nguyen & Duong (2021) suggested two approaches for handling imbalanced data for churn prediction. In the study, the authors compared the performance of two resampling methods which is SMOTE and Deep Belief Network and two-cost sensitive learning methods (focal loss and weighted loss) for churn prediction. The study results prove that focal loss and weighted loss methods are better than the resampling methods. The best results from the combination of XGBoost with focal loss and weighted loss were AUC of 66.18% and AUC of 65.92% on Cell2cell dataset.

Table 2.1: Summary of ML in CCP

Authors	**Architecture**	**Result**
Idris et al. (2012)	Chr-PmRF	AUC: 75.11%
Wu et al. (2021)	LR, DT, RF, NB, Multi- Layer Perceptron	AUC: Dataset1 LR: 91.34 Dataset2 RF: 91.40% Dataset3 RF: 58.66%
Nguyen & Duong (2021)	XGBoost with Focal Loss and Weighted Loss	AUC: Focal Loss: 66.18% Weighted Loss: 65.92%

2.3 Customer Churn Prediction in Ensemble Machine Learning

Hammoudeh et al. (2019) presented on churn prediction which focus on both individual and ensemble models. They explored various models including Artificial Neural Networks (ANN), Support Vector Machines (SVM), Convolutional Neural Networks (CNN), Averaging Ensemble Model (AEM), Random Forests (RF), and Selective Ensemble Model (SEM). The selective ensemble model which combines of a few different algorithms outperformed all of the individual models by at least 3% accuracy.

Wang et al. (2020) conducted research on the churn prediction. Their research conducted on the most widely used machine learning methods on churn prediction on a public dataset in telecom industry. They selected the top 4^{th} model which are LightGBM, XGBoost, Random Forest, and Decision Tree based on its performance. Then, these models were used to develop a soft voting technique ensemble model. The ensemble method improves the model performance which achieved the highest AUC over each individual model in their study.

Liu et al. (2022) proposed an ensemble system fully incorporating clustering and classification learning methods for customer churn prediction. Different classifiers including Gradient Boosted Tree (GBT), Decision tree (DT), Random Forest (RF), Deep Learning (DL), and Naive Bayes (NB) were ensemble into the model (k-medoids-GBT-DT-DL). The experiment involved three stages which are clustering, classification, and ensemble-based prediction. The proposed model captured recall 85.45%, f1-score 83.72% and accuracy 93.6%.

Table 2.2: Summary of Ensemble Learning in CCP

Authors	**Architecture**	**Result**
Hammoudeh et al. (2019)	CNN+ ANN+ SVM + RF	Acc: 93.90% Precision: 83.80% Recall: 84.60%
Wang et al. (2020)	LightGBM+ XGBoost+ Random Forest+ Decision Tree	AUC: 0.6890
Liu et al. (2022)	GBT + DT + +RF + Deep Learning + NB	Recall: 85.45% f1-score: 83.72% Accuracy: 93.6%.

2.4 Customer Churn Prediction in Deep Learning

In recent years, the rise of deep learning has revolutionized the field of churn prediction. Numerous deep learning approaches have been proposed on churn prediction. Previously, the research has done by using deep learning models such as convolutional neural networks and artificial neural networks and their variants.

2.4.1 Deep Neural Network

Albrecht et al. (2020) presented a deep learning method on churn analysis from a practical point of view. The churn prediction procedure includes pre-processing, modelling, and evaluation. The oversampling technique (SMOTE) is applied in the preprocessing step to deal with the imbalance data. In the modelling step, three distinct feedforward neural networks with one to three hidden layers were implemented for churn prediction. In the experiment, automatic grid search, and hyperparameters such as compositional details, regularizes of layer parameters, and dropout rate are investigated to improve prediction performance and prevent overfitting. The LIME algorithm is applied to interpret the deep learning churn classifier. The studies had also shown that there is an influence of pre-processing step and hyperparameter tuning. The result was increasing after these processes. The result obtained by the Neural Network model with 3 hidden layers was 0.645 AUC, 0.401 precision, and 0.484 recall.

In 2022, Wael Fujo et al. (2022) applied deep learning on customer churn prediction. In their study, deep back-propagation artificial neural network (Deep-BP-ANN) was implemented with Variance Thresholding and Lasso Regression feature selection methods. They worked on two popular telecom dataset which are IBM Telco and Cell2cell dataset. The model is implemented with early stopping to prevent overfitting. The best model was implemented with lasso regression for feature selection, early stopping, and large number of neurons on input and hidden layers, and regularization. Overall, the advancements in machine learning and deep learning techniques have played an important role in improving performance of churn prediction model in the telecommunication sector. Additionally, the adoption of ensemble learning methods has proven to be effective in further improving the performance of individual models which can led to more accurate and reliable churn prediction models.

Table 2.3: Summary of DNN in CCP

Authors	Architecture	Result
Albrecht et al. (2020)	DNN with 3 hidden layers	AUC: 0.645 Precision: 0.401 Recall: 0.484
Wael Fujo et al. (2022)	Deep-BP-ANN	Accuracy: 88.12%, Precision: 84.71% Recall: 93.05% F1-Score: 88.68%

2.4.2 Convolutional Neural Network

The development of Convolutional Neural Network has seen various of architecture that have being explored. CNN is popoluar for image recognition task. However, it can also be applied on the sequence data prediction such as churn predictions. Numerous CNN method has been explored on sequence data or sequence-to-image data on churn predictions.

Karanovic et al. (2018) presented on churn prediction models on the dataset provided by Orange telecommunication company. They apply the Convolutional Neural Network as a classifier on one-dimensional data for churn prediction. The pre-processing phase remove 30% of missing value, redundant data, and select relevant features using Lasso regression. In their experiment, several CNN model with different of complexities is developed. The best performance model consists of 1 convolutional layer with 512 filters, ReLu activation, MaxPooling, Flatten, and Dropout layers and output layer with sigmoid activation. The proposed model architecture shown in Figure 2.1. The model is evatualted using 10-fold cross-validation and obtained the accuracy of 98.85% with sensitivity of 86.11% and specificity of 98.97%.

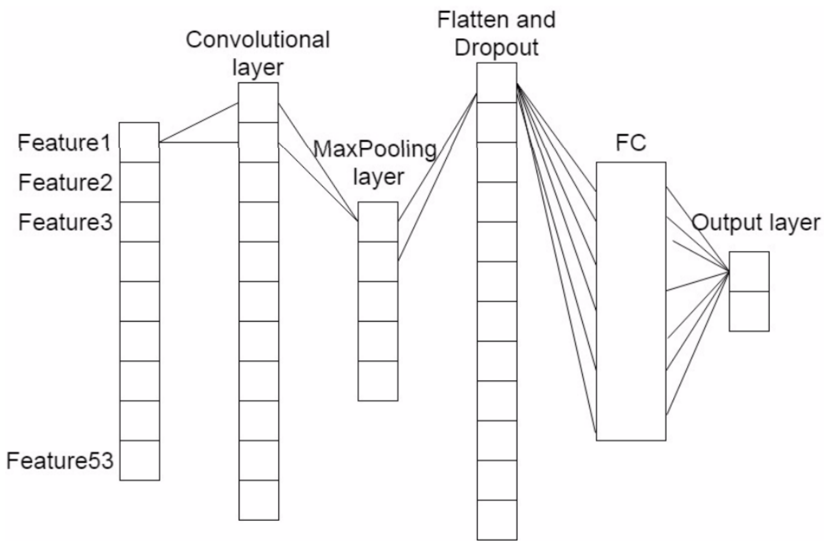

Figure 2.1. Model Architecture Design by Karanovic et al. (2018)

Ahmed et al. (2019) proposed a method "TL-DeepE" for churn prediction, using the principle of Transfer Learning and Ensemble-based Meta-Classification. The proposed model employed a multi-layer CNNs architecture and Genetic Programming and AdaBoost ensemble as a meta classifier. The author transformed the telecom data into two-dimensional. Three pre-trained CNNs models (AlexNet, Inception-ResNet-V2, and custom 6-layer deep neural-network model) and transfer learning were used to fine-tune on the selected dataset. The features extracted from the CNN models were then used as the input for the GP-AdaBoost ensemble classifier. The proposed model is evaluated using 10-fold cross-validation on common telecommunication prediction datasets which are Orange and Cell2cell. TL-DeepE model obtained prediction accuracy of 75.4% and 68.2%, and AUC values of 0.83 and 0.74 on the Orange and Cell2cell datasets, respectively.

Almufadi & Qamar (2022) studies on churn prediction in the telecom sector using deep learning algorithms such as Deep Neural Network, Convolutional Neural Network, and Recurrent Neural Network. However, this section specifically focuses on the CNN model proposed in the paper. The dataset is pre-processed on handling the imbalance nature. 1D CNN is proposed for churn prediction. The model consists of two convolutional layers with 32 filters and 1 kernel size. The polling, flatten, and dropout is adopted. The model is utilizing with early stopping to prevent overfitting.

Figure 2.2 shows the model architecture of their proposed CNN model. The result achieved accuracy of 91%.

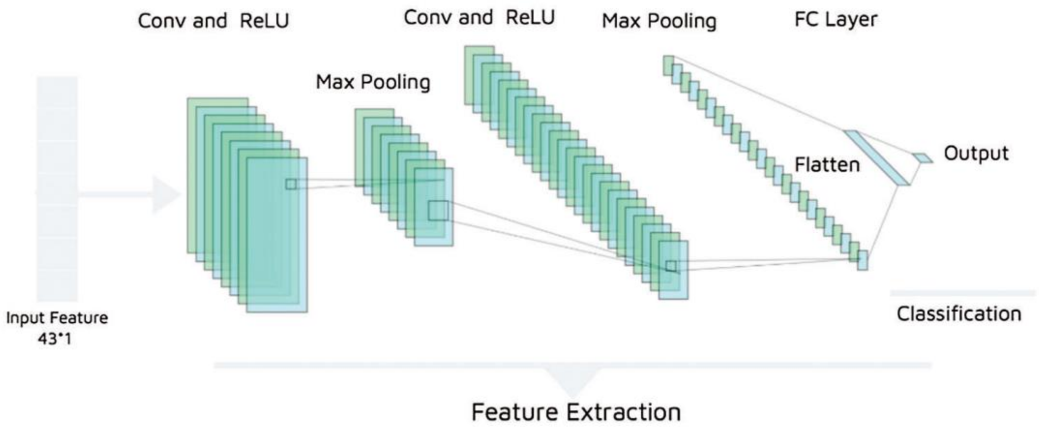

Figure 2.2. 1DCNN Model Architecture by Almufadi et al., (2022)

Usman et al. (2021) carried out a study on comparing the performance of various Deep Learning models on churn prediction. Two benchmark dataset is evaluated which are Cell2cell and KDD Cup. CNNs were used to process image data, while RNN is used to process sequence data. The CNNs is applied to train and test on one-dimensional and two-dimensional image. The deep learning architecture focus involves two main steps which are feature extraction and classification. In Figure 2.3, Architecture 1 and 2 is CNN based model. Several runs are implemented in the experiment. The result shows that the use of image data representation on Cell2cell dataset perform better than sequence data. KDD Cup achieve the best result on 2DCNN on the Architecture 1.

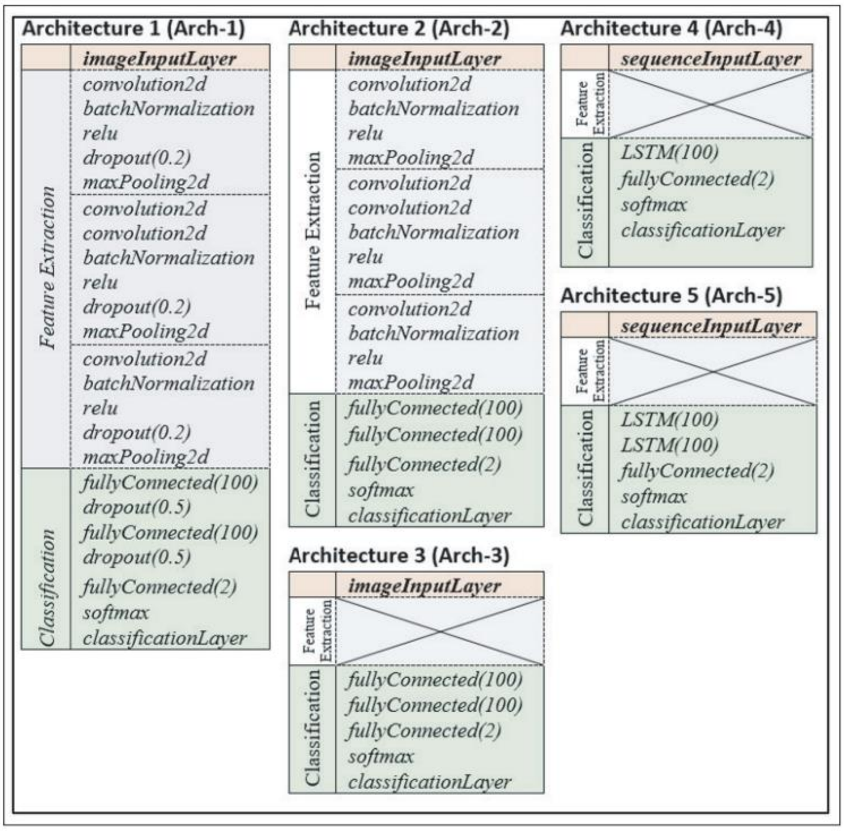

Figure 2.3. Model Architecture Design by Karanovic et al. (2018)

Furthermore, Tariq et al. (2022) proposed a model for predicting churn. The model adopted 2D convolution neural network. The proposed model consists several steps which is data loading, pre-processing, modelling using CNN, and finally churn prediction. The proposed model is train and evaluated on the IBM Telco dataset. The dataset is transformed into 2D data. Data loading is performed in parallel manner using Map-Reduce programming paradigm. Data balancing, standardization is done before modelling. Then, the processed data is fed to the 2D CNN layer to generate prediction. The model is trained with eight filters size and uses the ReLu activation, and an Apache Spark is used for parallel processing. Figure 2.4. shows the model design of the CNN. The proposed model achieves accuracy of 96.3% and a low training and validation loss of 0.004.

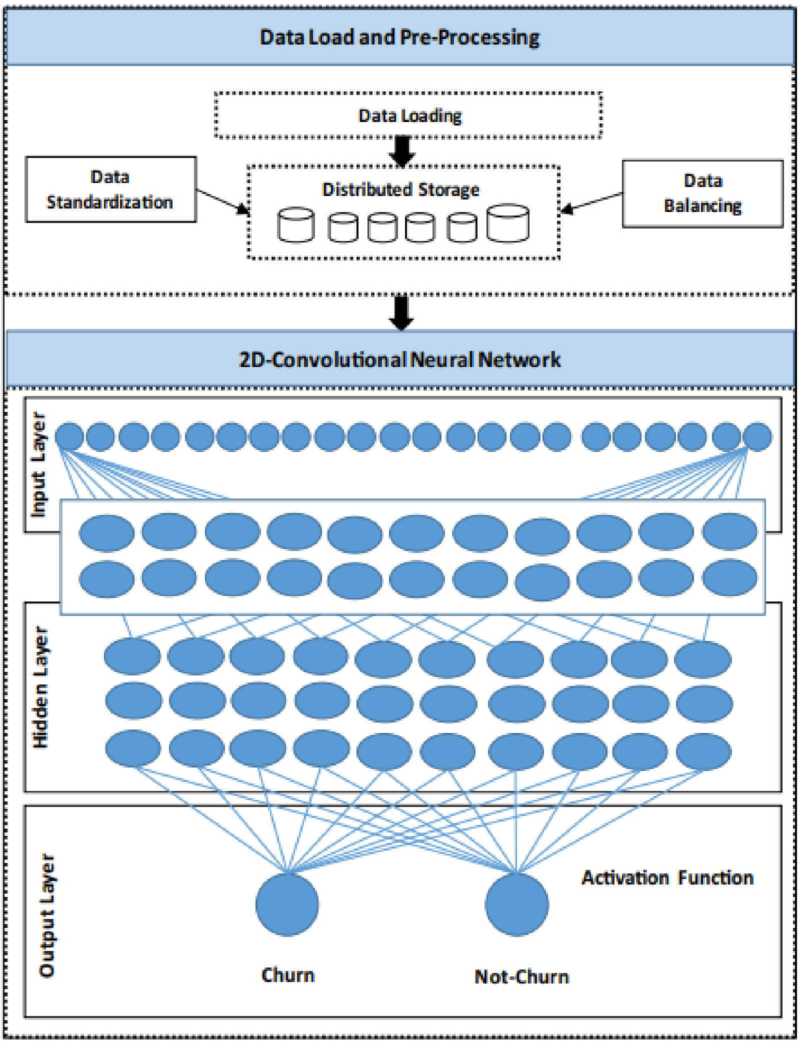

Figure 2.4. Model Design by Tariq et al. (2022)

Table 2.4: Summary of CNN in CCP

Authors	Architecture	Result
Karanovic et al. (2018)	1DCNN	Accuracy: 98.85% Sensitivity: 86.11% Specificity: 98.97%.
Ahmed et al. (2019)	TL-DeepE	Orange: Accuracy: 75.4% AUC: 83% Cell2cell:

		Accuracy: 68.2% AUC: 74%
Almufadi & Qamar (2022)	1DCNN	Accuracy: 91%
Usman et al. (2021)	Cell2cell: 1DCNN KDDCup: 2DCNN	Cell2cell Accuracy: 0.757 AUC: 83.2% Orange: Accuracy: 92.7% AUC: 70.8%
Tariq et al. (2022)	2DCNN	Accuracy: 96.3%

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this project, the research is aim to analyse the effectiveness of Convolution Neural Network on customer churn prediction. However, the application in churn prediction has not produced the optimal result. To enhance the prediction performance, this project proposed an ensemble method by ensemble CNN with the three other machine learning models. This chapter introduce the overview of proposed method in section 3.2. Subsequently, sections 3.3 – 3.5 describes the dataset selection and the pre-processing steps taken before the modelling steps. Additionally, sections 3.6 – 3.12 describe the selection of the individual model that constitute to the proposed model.

3.2 Proposed Method

The proposed work is structured into a few sections. Figure 3.1 depicts the design of the proposed framework. The first section focuses on the dataset adopted in the churn prediction model within the telecommunication industry. The subsequent section delves into Exploratory Data Analysis (EDA) and data pre-processing techniques. EDA helps in understanding the dataset, identifying patterns, and addressing missing values, irrelevant features, and data transformation. Data pre-processing involves cleaning the dataset, and handling missing values to prepare the data for modelling. The data transformation is carried out to transform the raw data into suitable format for machine learning. This process may involve data encoding and scaling the data to optimise model performance. As class imbalance is a common challenge in churn prediction, the next section focuses on handling imbalanced classes. Lastly, a churn prediction model is built and evaluated.

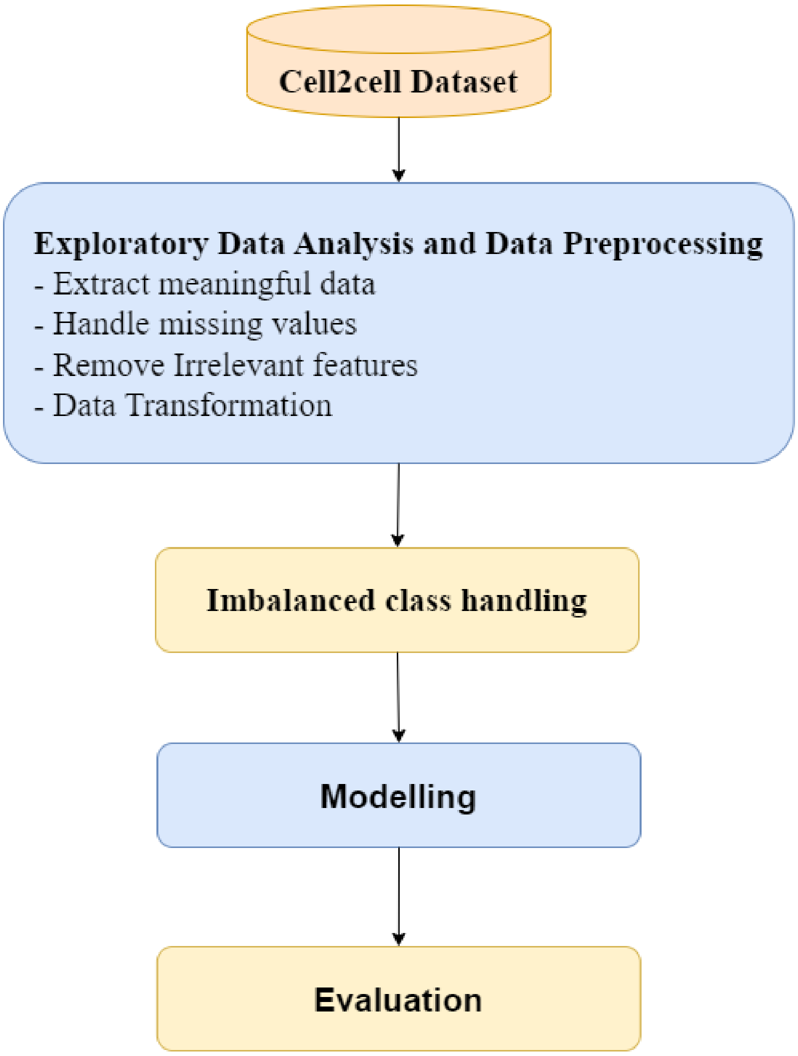

Figure 3.1. Design of Proposed Work

3.3 Dataset Preparation

In this study, the dataset is obtained from Kaggle which is an online data science community platform. It offers large amount of repository on publicly available datasets. The chosen dataset is obtained from Kaggle is Cell2cell. Cell2cell dataset consist a wide range of features, such as information on the customer demographics, patterns of usage, subscription to services, call information, and account characteristic. The diverse set of data serves as a valuable resource for study on churn prediction. Hence, Cell2cell is chosen in this study.

3.4 Exploratory Data Analysis and Data Pre-processing

Data pre-processing is very important before the modelling as the performance of the model closely depends on the quality of data. Exploratory Data Analysis plays an important role in data pre-processing. It provides insight and understanding of the dataset. EDA can help in identifying the missing value, dropping the irrelevant variables, and data transformation to effectively pre-process the data.

3.4.1 Missing Value Handling

In order to process the modelling, handling missing values is one of the crucial steps. Figure 3.2 visualises the features with missing values. There are 14 features in total that contain missing value such as Monthly Revenue, Monthly Minutes, Total Recurring Charge, etc. The features with missing value were imputed with the value of 0.

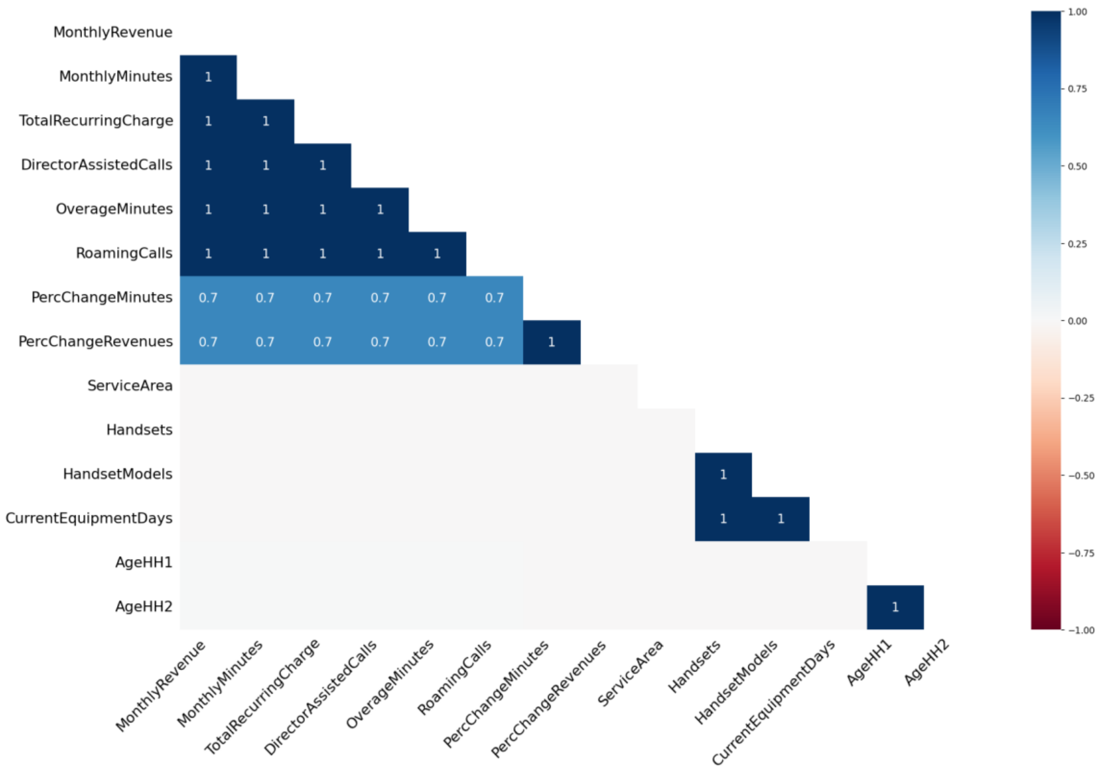

Figure 3.2. Features with missing values

3.4.2 Remove Irrelevant Features

During the exploratory data analysis (EDA), it was determined that the "Customer ID" and "Service Area" features do not contribute meaningful insights to the analysis. These features were found to lack significant relevance to the modelling. Thus, these 2 features are dropped from the dataset. Removing the irrelevant features can improve the quality of the dataset, reducing computational complexity, and improving model performance.

3.4.3 Data Transformation

Data Transformation techniques can significantly enhance the overall performance of the model (Mohammad et al., 2019). Categorical variable transformation is indeed crucial for the majority of machine learning models because most of the models typically only can deal with numeric values. Therefore, the categorical features will be transformed to numeric machine-friendly form before modelling. Label Encoding method is used to transform the categorical data in this study. For instances, consider the feature ‘Churn’ which contains two classes (Yes, No) transformed into 0, and 1; the categorical features with more than 2 classes ‘PrizmCode’ with 4 classes (Rural, Suburban, Town, Others) will be transformed into 0, 1, 2, and 3. Figure 3.3 shows the example of the Label Encoder. Moreover, data scaling is also applied in this study. Data scaling is a necessary process to standardise or normalise the different features to a specific range or scale which can be advantageous for numerous machine learning algorithms. This study used the common data scaling method min-max scaling to scaling features to similar range (0-1). Figure 3.4 and figure 3.5 shows the data before and after Data Scalling.

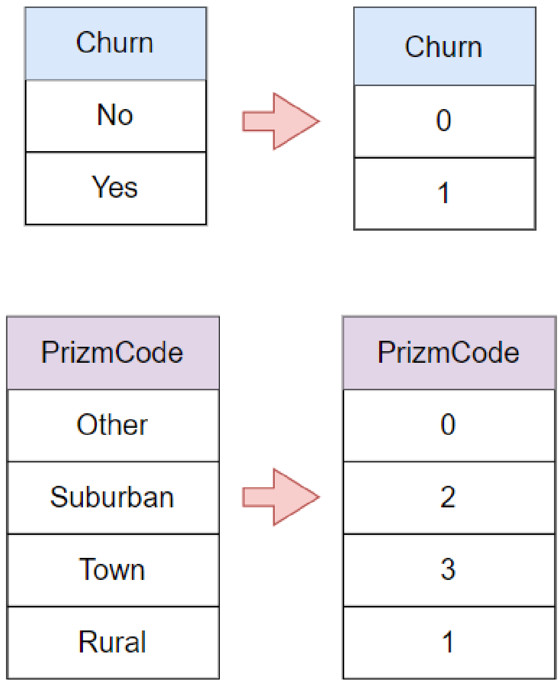

Figure 3.3. Example of Label Encoder

	Churn	MonthlyRevenue	MonthlyMinutes	TotalRecurringCharge	DirectorAssistedCalls	OverageMinutes
0	1	3.439456	5.393628	3.526361	0.223144	0.0
1	1	3.184698	2.397895	3.367296	0.000000	0.0
2	0	3.810433	2.197225	3.912023	0.000000	0.0
3	0	4.493680	7.180070	4.465908	0.806476	0.0
4	1	3.190888	0.000000	3.367296	0.000000	0.0

Figure 3.4. Data before Data Scaling

	Churn	MonthlyRevenue	MonthlyMinutes	TotalRecurringCharge	DirectorAssistedCalls	OverageMinutes
0	1.0	0.483394	0.605766	0.585675	0.043947	0.0
1	1.0	0.447590	0.269311	0.559256	0.000000	0.0
2	0.0	0.535533	0.246773	0.649727	0.000000	0.0
3	0.0	0.631559	0.806404	0.741719	0.158830	0.0
4	1.0	0.448460	0.000000	0.559256	0.000000	0.0

Figure 3.5. Data after Data Scaling

3.5 Imbalanced Class Handling

Customer churn is presented as a binary classification problem. Binary classification often encounters the imbalance class problems in which the churners belong to the minority class and non-churners belong to the majority class. Imbalance class problems can bring a huge negative impact on the model. It will lead the model biassed towards the majority class. The Cell2cell dataset had the similar problem, in which the data is highly imbalanced. Figure 3.7 shows the original Cell2cell churn rate with 71.2%, 23,238 of non-churners (label 0.0, cyan area) and only 28.8%, 9431 of churners (label 1.0, magenta area). Therefore, it is important to balance the class to prevent bias models. (Wu et al., 2021) utilise the imbalance class handling method include Synthetic minority oversampling technique (SMOTE), random oversampling (ROS), etc. Their work showed that the SMOTE method stands to be most relevant for data balancing. Strategy employed by SMOTE involves generating synthetic instances of the minority class. The SMOTE technique is obtained from the imblearn library. The SMOTE algorithm is used to address the class imbalance by oversampling the minority of classes. Consider the example shown in Figure 3.6. (Nguyen & Duong, 2021).

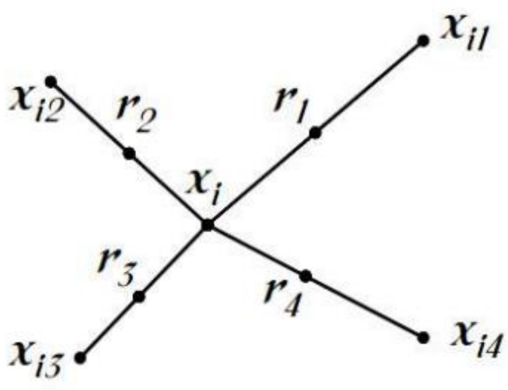

Figure 3.6. Synthetic Minority Oversampling Technique (SMOTE) (Nguyen & Duong, 2021)

An instance x_i from the minority class is selected. Then, several nearest neighbours of the same class, represented as x_{i1} to x_{i4}, are chosen using a distance measure. A randomised interpolation is then performed, and result in the generation of new instances labelled as r_1 to r_2. Hence, SMOTE is adopted for this study. As shown in Figure 3.8, the distribution of churners and non-churners are balanced which result in becoming 50% for each class after applying the SMOTE. The application of SMOTE increased the number of churners from 9,431 to 23,238.

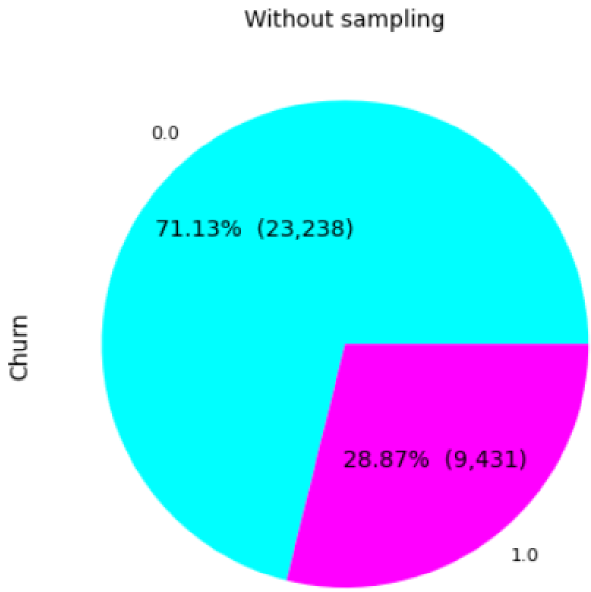

Figure 3.7. Percentage of churners and non-churners without sampling

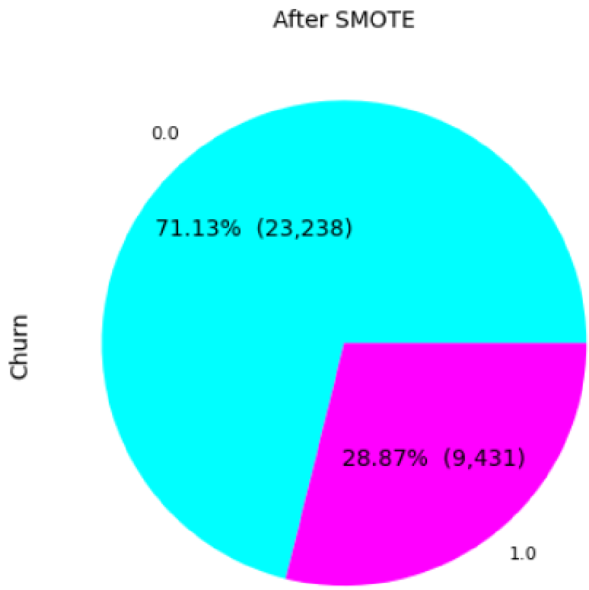

Figure 3.8. Percentage of churners and non-churners after SMOTE

3.6 Ensemble Learning

Ensemble learning is a technique that combines the multiple machine learning algorithms' predictive output and aims to improve the performance of the single classifier. In ensemble learning, multiple models were integrated into a meta-classifier to eliminate limitations of each algorithm and enhance the generalisation potential. Ensemble classifiers can be further classified into two types: homogeneous and heterogeneous ensembles. Homogeneous ensembles use the same combined base classifiers but adopt different sampling methods, while heterogeneous ensembles combine multiple different types of classifiers or models to make predictions (Idris et al., 2012).

Given the complexity of customer churn prediction, which involves a large amount of customer data, it is recommended to use heterogeneous ensembles because they leverage the strengths of different base classifiers compared to homogeneous ensembles. Heterogeneous ensembles can capture diverse aspects of customer data, thereby enhancing performance. Therefore, heterogeneous ensembles are deemed to be more suitable for customer churn prediction. Not only that, heterogeneous ensembles have also received limited attention in literature on customer churn prediction, the proposed model in this paper employed the heterogeneous ensemble for customer churn prediction.

Heterogeneous ensembles combine multiple base classifiers which is from the single classifier or homogeneous ensembles. Each base classifier makes prediction independently and transfer the prediction as an input into the meta-learner. The meta-learner learns to integrate the predictions to generate the final prediction. Heterogeneous ensemble takes different perspective from different base classifiers and increases the diversity by complement on the strength and mitigate the weakness to improve the accuracy of the model.

Figure 3.9 shows the proposed stacking ensemble model for churn prediction. The stacking ensemble is comprising with two levels which is base-classifiers and meta-learner. In the base classifier, the pre-processed data is employed to train models and make predictions. Then, the meta-learner is adopted to map the output of the base classifier to the corresponding actual label. This ensemble approach is employed by the Nexflix team known as “The Ensemble”, achieved the comparable accuracy in the wining team submission (Sill et al., 2009). It is crucial to ensure that the ensemble model has a wide diversity, this study adopted various base classifier which are Logistic Regression, Decision Tree, Support Vector Machine (SVM) and Convolutional Neural Network (CNN) models. The diverse selection of classifier is to produce a higher quality and reliable model. The CatBoost classifier is selected as the meta-learner stems from its capability to effectively handle diverse data to guarantee the performance of the ensemble model.

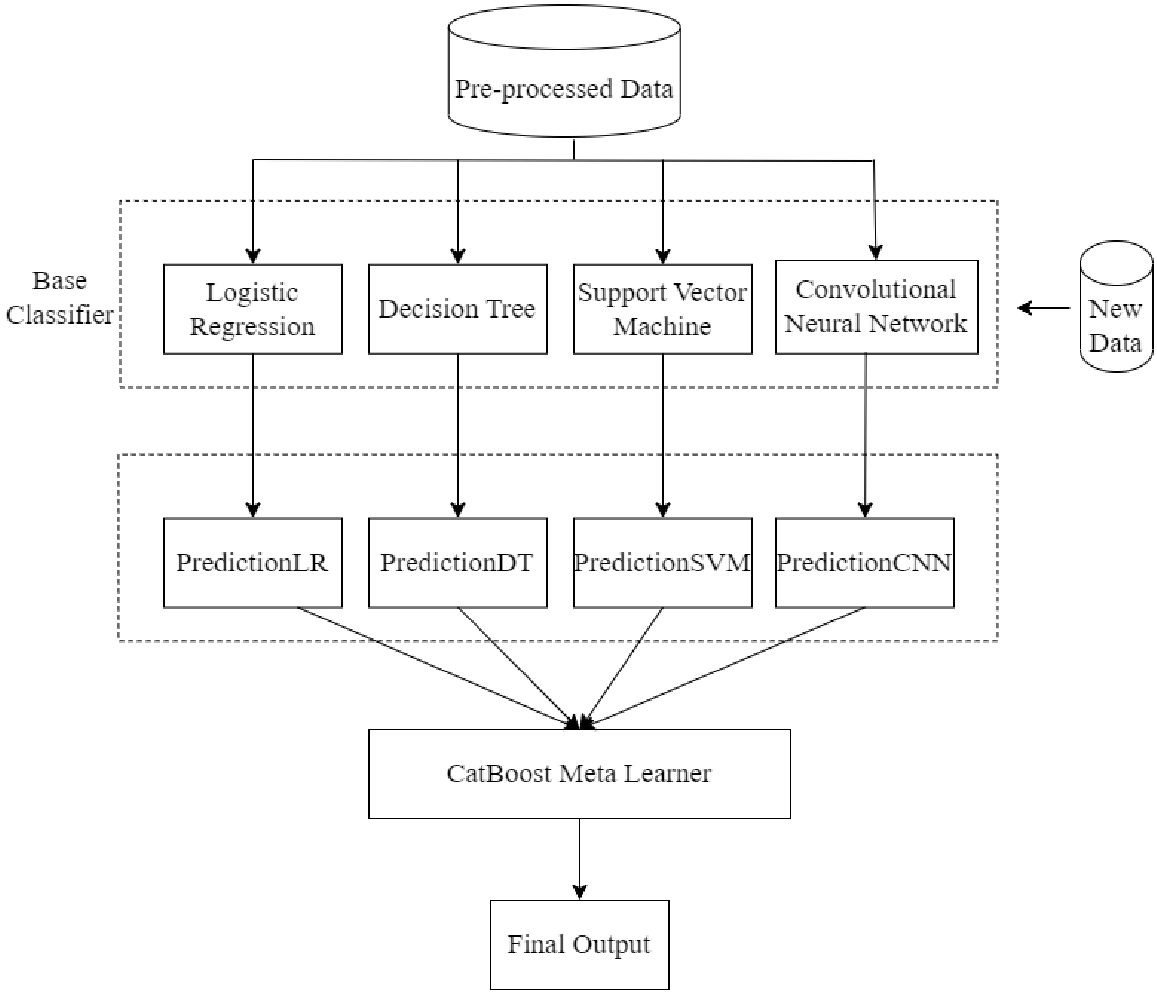

Figure 3.9. The Proposed Ensemble Model

3.7 Logistic Regression

Logistic regression is a widely adopted statistical modelling technique that falls under the category of supervised machine learning algorithms. It can be binary, multinomial or ordinal. In this study, binary logistic regression is being adopted. The purpose of logistic regression is to use the real-valued input to a range between 0 and 1. In logistic regression, if the probability of prediction is more than 0.5, output will be assigned to class 0 (non-churners), else the output is assigned to class 1 (churners). The logistic regression equation can be written as:

$$P = \frac{1}{1 + e^{-z}} \tag{3.1}$$

where P is probability of churn and z represents the linear combination of customer's features weighted by their coefficient which can be written as:

$$z = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n \tag{3.2}$$

where $\beta_0, \beta_1, \beta_2, \ldots, \beta_n$ are coefficients to the independent variables $X_1, X_2, \ldots$, X_n.

3.8 Decision Tree

Decision tree is a supervised machine learning method that is used to solve classification and prediction problems. It consists of a tree-like model with the internal node, branches, and leaf nodes. The interval node represents a test on feature, branch corresponds to the possible outcome of the test, and leaf nodes indicate the final class label. It is used for making decisions based on the input from the dataset and leads to the prediction of the target. The construction of a decision tree typically operates in a top-down manner, in which the tree is constructed and evaluated from the root node to downwards. The root node indicates the initial feature that is used for classification and the following nodes are selected based on the criteria such as information gain or impurity. Entropy refers to one of the several impurity measures used in decision trees.

$$Entropy(E) = -\sum_{i=1}^{N} p_i \, log_2 \, pi \tag{3.3}$$

where pi is the probability of each class i within the node.

Decision Tree usually involves two main phrases which are tree building and tree pruning. In tree building phrases, the algorithm recursively partitioning on the train data based on the attribute value. Data partitioning process is done until the partitions contain mostly identical values. The pruning step is to focus on select and remove branches that contain noisy data with the largest estimated error rate. This results in a more optimal model.

3.9 Support Vector Machines (SVM)

Support vector machine is a supervised learning algorithm that is used for the classification for both linear and nonlinear problems. It involves transforming the data into a higher dimensional space to identify a hyperplane to separate the data []. The chosen hyperplane which is known as the support vector maximises the margin which is the distance between the hyperplane and the nearest data points from each class. SVM in binary classification performs to find an optimal hyperplane for separating the data points of two different classes. The function can be defined as (Ben Jabeur et al., 2021):

$$Z = f(y) = sign\left[\sum_{i=1}^{N} y_i d_i K(x, x_i) + c\right] \tag{3.4}$$

where $sign$ is the sign function, p_i and c are parameter that refers to hyperplane. $K(x, x_i)$ denotes the kernel radial basis function (RBF).

3.10 CatBoost

CatBoost is a gradient boosting on decision tree algorithm that is well-suited for handling binary classification tasks. CatBoost stores the binary features in a continuous vector B, then the values of leaf nodes in decision trees stored in a 2-dimensional float number vector. To establish the binary vector for sample y (Deng et al., 2021):

$$\sum_{i=0}^{d-1} 2^i \cdot B(y, f(t, d)) \tag{3.5}$$

where the binary vector $B(y, f)$ reads the value of binary feature f for sample y. Similarity, $f(t, d)$ retrieves the number of binary features at depth d in tree t. CatBoost constructs vectors in parallel with data, and results in faster processing while the automatic handling of categorical features.

3.11 Convolutional Neural Network

Convolutional neural network is a deep learning method that belongs to the type of artificial neural. CNN can take images or features as an input which is either 1D, 2D or 3D matrix. CNN is most commonly used in processing image; however, CNN can also be used in processing one-dimensional data such as the churn prediction in telecom industry. CNN model typically consists of several steps. Firstly, the convolutional step involve in extracts the important features from the customer data while preserve the relationship between features and label. This is done through the use of filters and convolutional operation. Then, the non-linear activation functions like rectified linear unit (ReLu) and sigmoid are used to establish the connections between input features and the hidden layers. These enable the CNN model to learn the complex relationship in the data. It is important to determine the number of hidden layers to increase the performance of CNN. Adding more hidden layers might enhance the ability of model on distinguish the pattern for each class. Finally, the model classification steps occur. The fully connected layers taking the learned features as an input to perform classification task, and generate predicted output.

In short, the core architecture of CNN based churn prediction model is consists of an input layer, hidden layer, and an output layer. The input layer receives customer information, and the hidden layer with calculated weights by using the sigmoid function and connect the input to the class label.

$$f(x; y, z) = x_1 y_1 + x_2 y_2 + \cdots + x_n y_n + z \tag{3.6}$$

where $f(x; y, z)$ is depends on the value of the input vector $x_1, x_2, \ldots, x_n$, z represent co-efficient, and value of weights $y_1, y_2, \ldots, y_n$.

The sigmoid is utilized to maps the predicted values in the range of 0 to 1 as:

$$f(x; y, z) = g(y^T x + z), \text{ where} \quad (3.7)$$

$$g(j) = \frac{1}{1+exp^{-j}} \quad (3.8)$$

Then, the churn prediction model can be designed as:

$$f\left(x^{(1)}; y, z\right) \approx a^{(1)} \ldots \quad (3.9)$$

$$\ldots \ldots \ldots$$

$$f\left(x^{(n)}; y, z\right) \approx a^{(n)} \ldots \quad (3.10)$$

where $f\left(x^{(1)}; y, z\right) \approx a^{(1)}$ denotes the label for first customer, and $f\left(x^{(n)}; y, z\right) \approx a^{(n)}$ is denotes as n^{th} customer label.

3.12 Variational Autoencoder

The variational autoencoder was first introduce by Knigma and Welling in 2013 (Kingma & Welling, 2013). VAE is a neural network that is using unsupervised learning and data generation. It is part of autoencoders, in which a neural network that is designed for reduce the dimension of data and reconstruct the input data. The different between VAE and autoencoders is that VAE using a probability distribution to represent latent vector, while autoencoder is using function to represent latent vector (Zilvan et al., 2022).

Figure 3.10 shows the VAE architecture. The two main parts of VAE are encoder and decoder. The encoder take part in using input datapoint and maps the latent space representation. In encoding process, the convolutional or fully connected will be involve to extract the meaningful data. The encoder outputs of points that capture the probabilistic of latent vector. The distribution commonly represents by mean and standard deviation. It can be denoted as $q(z|x)$, where z represent latent vector, and x denotes input data. The decoder learnt from the latent space and reconstruct the original input data. The probability distribution $p(x|z)$ used by decoder to reconstruct output x given latent vector z in the latent space.

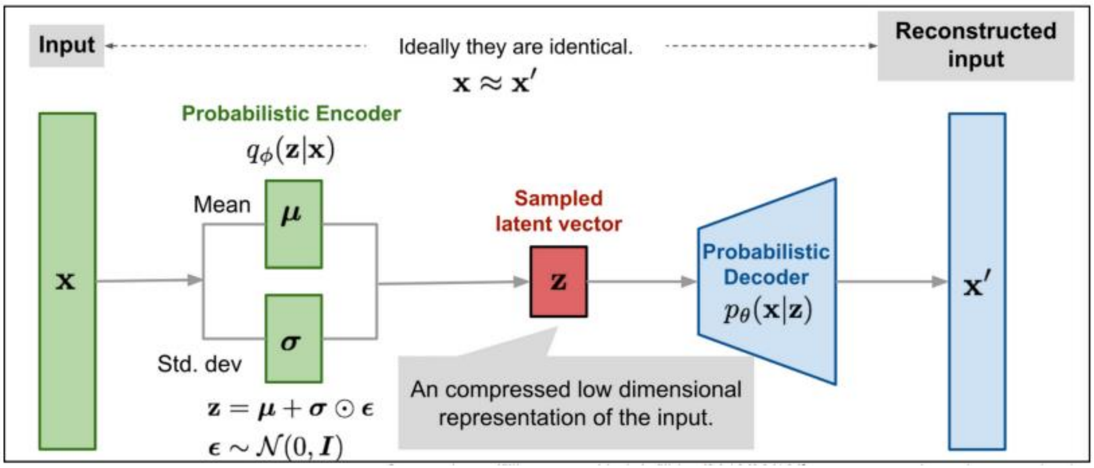

Figure 3.10. VAE Architecture (From Autoencoder to Beta-VAE | Lil'Log, accessed June 6, 2023)

CHAPTER 4

EXPERIMENTAL AND ANALYSIS

4.1 Dataset

The dataset adopted in this study is the Cell2cell dataset which consists of 51,047 instances and 58 attributes. The dataset used in the proposed model is openly available and collected by the Teradata Center at Duke University. It can be accessed and downloaded from Kaggle (Telecom Churn (Cell2cell) | Kaggle, accessed May 29, 2023). The dataset is selected based on their popularity on the study of churn prediction. Table 4.1 describes the details of Cell2cell data.

Table 4.1: Details of the Cell2cell dataset.

#	Numerical Features	Data Format	Description	Type
1	CustomerID (Discarded)	*Int*	Customer ID	*Numerical*
2	Churn	*(Yes/No)*	Whether the Customer churned or not	*Categorical*
3	MonthlyRevenue	*float*	Monthly Revenue	*Numerical*
4	MonthlyMinutes	*float*	Monthly minutes used	*Numerical*
5	TotalRecurringChar ge	*float*	Total Recurring Charge	*Numerical*
6	DirectorAssistedCall s	*float*	Number of director assisted calls	*Numerical*
7	OverageMinutes	*float*	Overage minutes of use	*Numerical*
8	RoamingCalls	*float*	Number of roaming calls	*Numerical*
9	PercChangeMinutes	*float*	% change in minutes of use	*Numerical*
10	PercChangeRevenue s	*float*	% change in revenues	*Numerical*
11	DroppedCalls	*float*	Number of dropped voice calls	*Numerical*
12	BlockedCalls	*float*	Number of blocked voice calls	*Numerical*
13	UnansweredCalls	*float*	Number of unanswered voice calls	*Numerical*

14	CustomerCareCalls	*float*	Number of customer care calls	*Numerical*
15	ThreewayCalls	*float*	Number of three way calls	*Numerical*
16	ReceivedCalls	*float*	Unrounded received voice calls	*Numerical*
17	OutboundCalls	*float*	Number of outbound voice calls	*Numerical*
18	InboundCalls	*float*	Number of inbound voice calls	*Numerical*
19	PeakCallsInOut	*float*	Number of in and out peak voice calls	*Numerical*
20	OffPeakCallsInOut	*float*	Number of in and out off-peak voice calls	*Numerical*
21	DroppedBlockedCalls	*float*	Number of dropped or blocked calls	*Numerical*
22	CallForwardingCalls	*float*	Number of call forwarding calls	*Numerical*
23	CallWaitingCalls	*float*	Number of call waiting calls	*Numerical*
24	MonthsInService	*int*	Months in service	*Numerical*
25	UniqueSubs	*int*	Number of unique subscription	*Numerical*
26	ActiveSubs	*Int*	Number of active subscription	*Numerical*
27	ServiceArea (Discarded)	*string*	Communications service area	*Categorical*
28	Handsets	*float*	Handsets issued	*Numerical*
29	HandsetModels	*float*	Handset models issued	*Numerical*
30	CurrentEquipmentDays	*float*	Number of days of the current equipment	*Numerical*
31	AgeHH1	*float*	Age of first HH member	*Numerical*
32	AgeHH2	*float*	Age of second HH member	*Numerical*
33	ChildrenInHH	*(Yes/No)*	Presence of children in HH	*Categorical*
34	HandsetRefurbished	*(Yes/No)*	Whether the handset is refurbished or not	*Categorical*
35	HandsetWebCapable	*(Yes/No)*	Whether the handset is web capable or not	*Categorical*
36	TruckOwner	*(Yes/No)*	Whether the subscriber owns a truck	*Categorical*
37	RVOwner	*(Yes/No)*	Whether the subscriber owns a recreational vehicle	*Categorical*
38	Homeownership	*(Known Unknown)*	Whether the home ownership is missing	*Categorical*

39	BuysViaMailOrder	*(Yes/No)*	Whether the subscriber buys via mail order	*Categorical*
40	RespondesToMailOffers	*(Yes/No)*	Whether the subscriber responds to mail order	*Categorical*
41	OptOutMailings	*(Yes/No)*	Whether the subscriber has chosen not to be solicited by mail	*Categorical*
42	NonUSTravel	*(Yes/No)*	Whether the subscriber has travelled to a non-US country	*Categorical*
43	OwnsComputer	*(Yes/No)*	Whether the subscriber owns a computer or not	*Categorical*
44	HasCreditCard	*(Yes/No)*	Whether the subscriber owns a credit card or not	*Categorical*
45	RetentionCalls	*int*	Number of calls made to retention team	*Numerical*
46	RetentionOffersAccepted	*int*	Number of retention offers accepted	*Numerical*
47	NewCellphoneUser	*(Yes/No)*	Known to be a new cell phone user	*Categorical*
48	NotNewCellphoneUser	*(Yes/No)*	Known not to be a new cell phone user	*Categorical*
49	ReferralsMadeBySubscriber	*int*	Number of referrals made by subscriber	*Numerical*
50	IncomeGroup	*int*	Income group	*Numerical*
51	OwnsMotorcycle	*(Yes/No)*	Whether the subscriber owns a motorcycle or not	*Categorical*
52	AdjustmentsToCreditRating	*int*	Number of adjustments to customer credit rating	*Numerical*
53	HandsetPrice	*string*	Handset price	*Categorical*
54	MadeCallToRetentionTeam	*(Yes/No)*	Whether the subscriber has made call to retention team	*Categorical*
55	CreditRating	*String*	The credit rating of the subscriber	*Categorical*
56	PrizmCode	*(Rural/Suburban/Town/Other)*	The prizm code of the subscriber	*Categorical*
57	Occupation	*(Clerical/ Crafts/ Homemaker/Professional/Self/ Student/Retired/ Other)*	The occupation of the subscriber	*Categorical*
58	MaritalStatus	*(Yes/No/Unknown)*	The marital status of the subscriber	*Categorical*

4.2 Performance Metrics

In churn prediction, several performance metrics can be used to assess the performance and effectiveness of the churn prediction model. The most used performance metric was discussed. First, the performance metrics are accuracy. The accuracy is commonly used to see the overall performance of the model. The accuracy is calculated by (4.1), the number of correct classified data divided by the total numbers of data. The precision and recall are also adopted in the study. Precision (4.2) is the measure of correctly classified data (true positive) out of the total number of data whereas the recall (4.3) measures the correctly classified data out of all actuals churn data. Since the data is imbalanced, the F1-score and Area Under Curve (AUC) is also included. F1-score (4.4) combines the precision and recall, then provides a balanced assessment for performance of the model. The AUC curve (4.5) illustrates how well the binary classifier model can distinguish between positive and negative instances.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.2}$$

$$Recall = \frac{TP}{TP + FN} \tag{4.3}$$

$$F1 = \frac{Precision \times Recall \times 2}{Precision + Recall} \tag{4.4}$$

$$AUC = \int_{-\infty}^{\infty} y(t)\, dx\, (t) \tag{4.5}$$

Moreover, confusion matrix also being adopted to measure the performance of the model. It provides summary of the predicted outcome compared to the actual outcomes from the models. The figure of confusion matrix is shown in figure 4.1.

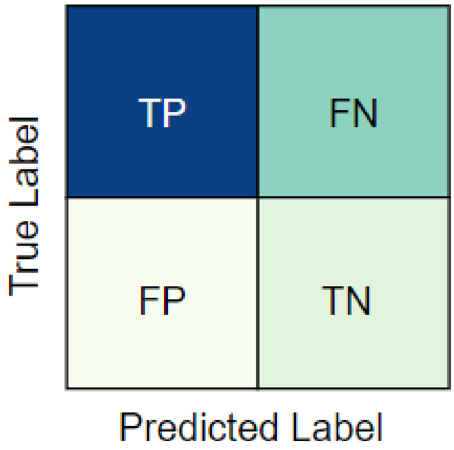

Figure 4.1. Confusion matrix

where the following describe each term:

TP is true positive, where a customer is churn (positive) and classified as churn (positive);

TN is true negative, where a customer is not churn (negative) and classified as not churn (negative);

FP is false positive, where a customer is not churn (negative) and classified as churn (positive);

FN is false negative, where a customer is churn (positive) and classified as churn (negative).

4.3 Model and Parameter Configuration

The ensemble model was developed using the two popular python libraries. The three base classifier such as Logistic Regression, Decision Tree, Support Vector Machine were imported from the scikit-learn library which is a python library that offers the implementation of machine learning and statistical models. The CNN model is developed using the Tensorflow and Keras. Tensorflow is library for numerical computation, while Keras is the high-level neural networks API runs on top of the Tensorflow. The ensemble meta learner is using CatBoost, a gradient boosting library for handling category features in machine learning models.

Additionally, this experiment was carried out using Jupyter Notebook with Python 3.10 in Google Colaboratory. Google Colab is a cloud-based platform which offers powerful hardware accelerators which can speed up the deep learning training. The computational resources employed for this experiment included Intel Core i5 processors equipped with 4 GB of RAM. The details of model training set up is described in this section.

4.3.1 Logistic Regression

Six hyperparameter were used to optimized the performance of LR. The first parameter is C which is the inverse of regularization strength. C is use for determine the amount of regularization to the model. The second is penalty which is to specifies the type of regularization applied to the model. The third parameter is solver which is used for optimization. The max iterations (max_iter) which is to specify the maximum number of iteratiobns for the solver to converge. Subsequently, random state (random_state) and class weight (class weight) is used to initialize the random number generator to a fixed state, and assign the weight to different classes to handle the imbalance class problem. Table 4.2 describe the chosen hyperparameter and best parameter.

Table 4.2: Parameter Configuration for LR

	Hyperparameter	**Best Parameter**
C	0.01, 0.1, 1.0	1.0
penalty	l1, l2	l2
solver	liblinear, saga	Saga
max_iter	1000, 1500	1500
random_state	42	42
class_weight	balanced	balanced

4.3.2 Decision Tree

To optimize the model, eight hyperparameter were selected. The criterion parameter is modified to select the best attribute for splitting the data. Max depth (max_depth) is used to set the maximum depth or levels for decision trees. Max features (max_features) is used to determine the max number of features for best split in each node. The minimum samples split is used to specifies the minimum number of samples needed for splitting in interna node which helps in preventing overfitting. Splitter is set to select the splitting strategy for each node. Moreover, random state (random_state) and class weight (class weight) is used to initialize the random number generator to a fixed state, and assign the weight to different classes to handle the imbalance class problem. Table 4.3 specifies the details of parameter setting for decision tree.

Table 4.3: Parameter Configuration for DT

`

	Hyperparameter	**Best Parameter**
criterion	gini, entropy	entropy
splitter	best, random	best
max_depth	None, 5, 10	None
min_sample_split	2, 5, 10	2
min_sample_leaf	1, 2, 4	1
max_features	None	None
random_state	42	42
class_weight	balanced	balanced

4.3.3 Support Vector Machine

In SVM, three parameters play important role to control the performance of the SVM models. The first parameter is C which is used to control the trade-off between training error and also the margin width. Kernel parameter is to determines the type of transformation for non-linear classification. The third one is gamma. Gamma parameter is used to define the influence of shape for the decision boundary. The random_state and class_weight is also adopted in this model. Table 4.4 represents th details of hyperparameter setting for SVM model,

Table 4.4: Parameter Setting for SVM

	Hyperparameter	Parameter
C	0.1, 1.0, 10	10
kernel	linear, rbf	rbf
gamma	scale, auto	scale
random_state	42	42
class_weight	balanced	balanced

4.3.4 Convolutional Neural Network

The implementation of CNN was done in two phases. Firstly, the CNN was implemented with Variational Autoencoder (VAE). Then, the new samples generated by CNN with VAE that share similarities with the training data was used to train the CNN model. The model architecture of CNN with VAE will first describe and followed by CNN model.

The CNN with VAE is implemented with early stopping to prevent overfitting. Table 4.5 shows the CNN with VAE model configuration. Figure 4.2 represent the model architecture of CNN with VAE. Firstly, the input layer was initialize using the dimension of the dataset. Then, the input is passed to the two dense 1D convolutional layer with filter ad filter size, and padding is set to 'same' to preserve the input size. ReLU activation is applied in each convolutional layer. The output of the second convolutional layer is flattened. Next, the flattened output is then adopted to the hidden dense layer. Linear activation function is used in this hidden layer. After this, two dense layers are created to generate the mean and logarithm of the standard deviation of the latent space variables. Then, the encoder model maps the input to the mean of the latent space variables. Finally, the decoder with dense and convolutional layer were created to decode the output in output layer.

After getting the new samples of data, the CNN model is carried out to train and predict the data. The model configuration is show in Table 4.6 and model architecture is shown in Figure 4.3. The CNN model is implementing with early stopping to monitor the loss of the model. The model architecture is simple. It consists of one of 1D Convolutional layer, a fully connected layers with 256 units, flatten layers, and a fully connected with 1 unit and sigmoid function.

Table 4.5: Model Configuration for CNN with VAE

	Parameter
Input Dimension	(55,1)
Batch size	100
Number of filters	64
Kernel size	3
Hidden layer dimension	256
Latent space dimension	2
Callbacks	monitor = validation loss patience = 8
Number of epochs	200
Optimizer	Adam

Table 4.6: Model Configuration for CNN

	Parameter
Input dimension	(2,1)
Number of filters	32
Kernel Size	2
Callbacks	monitor = validation loss patience = 3
Number of epochs	50
Loss Function	Binary crossentropy
Optimizer	Adam

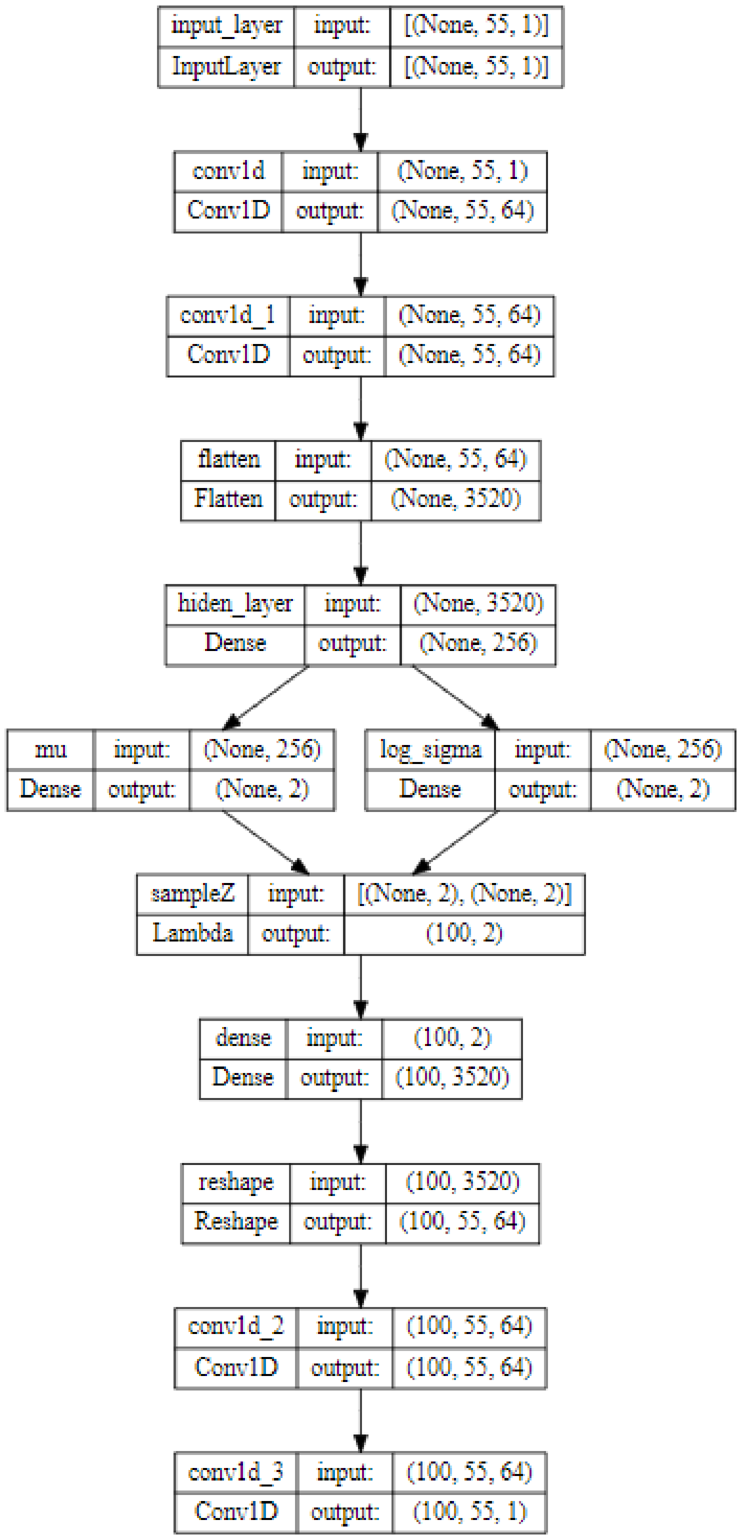

Figure 4.2. Model architecture for CNN with VAE

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d_6 (Conv1D)           (None, 1, 32)             96

 dense_2 (Dense)             (None, 1, 256)            8448

 flatten_1 (Flatten)         (None, 256)               0

 dense_3 (Dense)             (None, 1)                 257

=================================================================
Total params: 8,801
Trainable params: 8,801
Non-trainable params: 0
_________________________________________________________________
```

Figure 4.3: Model architecture of CNN

4.3.5 CatBoost

Hyperparameter setting is also applicable to meta-learner CatBoost. There were three key hyperparameters that are commonly used in the CatBoost. The first is iteration which is used to determine the number of iterations of boosting in the ensemble. Parameter depth is used to set the maximum deptshs of each decision tree in an ensemble. Learning rate (learning_rate) was set to control the step size during the gradient boosting. The random_state and auto_class_weight is also adopted in this model. Table 4.7 presented the details of the hyperparameter setting is describe.

Table 4.7: Parameter Configuration for CatBoost

	Hyperparameter	**Best Parameter**
iterations	50. 100, 150	50
depth	4, 6, 8, 10	4
learning_rate	0.1, 0.01	0.01
random_state	42	42
auto_class_weights	Balanced	Balanced

4.4 Result and Analysis

4.4.1 Area Under Curve (AUC) on Base Classifier

AUC score can be uses to evaluate the performance of model on both positive and negative classes on the tested data. A higher score can indicate the better overall performance of the model. The obtained AUC from each base classifier is graphically shown in Figure 4.4. Based on the result, the logistic regression classifier obtained the highest AUC score of 62.00%, which indicate that it can perform relatively better in discriminant between churn and non-churn customer. The SVM classifier obtained the second highest AUC score of 59.00%, while the decision tree and CNN classifiers showed lower AUC scores of 54.00% and 53.00%, respectively.

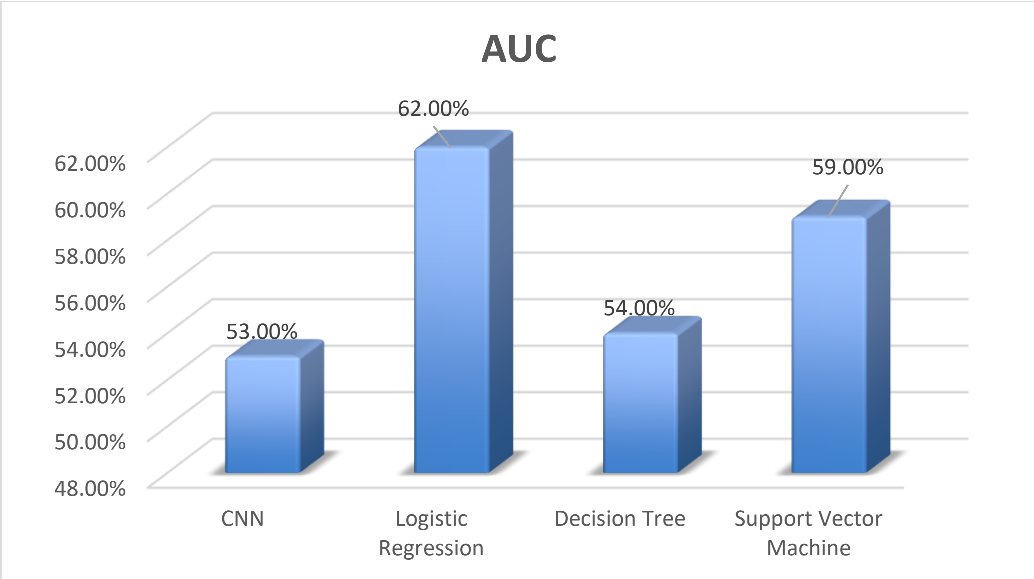

Figure 4.4. AUC of Base Classifier

4.4.2 Performance Analysis on Base Classifier

In this study, multiple base classifiers were tested on the pre-processed data such as Convolutional Neural Network, Logistic Regression, Decision Tree, Support Vector Machine. Figure 4.5 graphically presented the results obtained in which accuracy, precision, recall, f1 score, AUC.

The logistic regression classifier has the better overall performance. It achieves a better AUC, accuracy, precision, recall, and F1-score at 62.00%, 59.65%, 66.61%, 59.65%, and 61.56% respectively. However, the CNN classifier performs the lowest performance among the base classifiers in all metrics. CNN obtained lowest AUC, accuracy, recall, and F1-score. Although the precision is the highest among the base classifiers, it is still relatively low. The decision tree classifier performs reasonably well in terms of accuracy, precision, recall, and F1-score. However, it shows a relatively low AUC at 54%. The SVM classifier performs similarly to the logistic regression model in terms of accuracy, precision, recall, and F1-score. However, it achieves a lower AUC at 59% which is 3% lower than AUC of LR. Hence, the LR base classifier gives the most significant result.

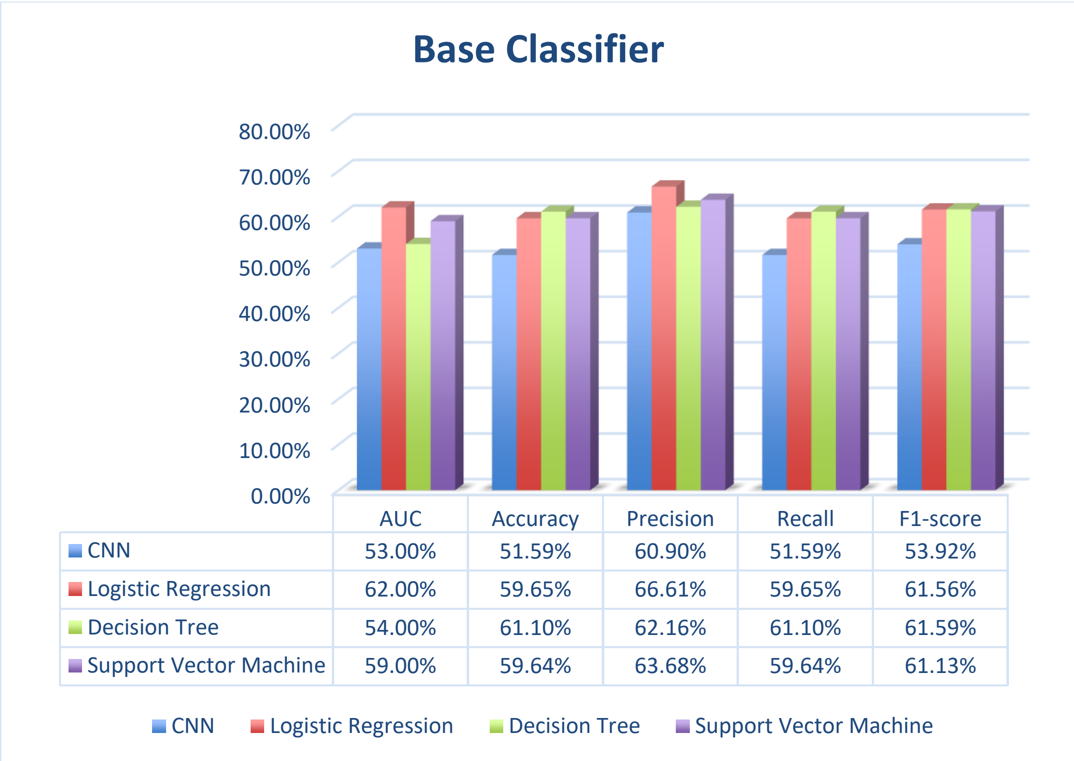

Figure 4.5. Performance Analysis of Base Classifier

4.4.3 Performance Analysis on Base Classifier

The proposed ensemble is developed by ensemble three machine learning methods, Logistic Regression, Decision Tree, and Support Vector Machine (SVM), and along with deep learning method which called Convolutional Neural Network

(CNN). The ensemble model is trained and tested using the pre-processed data. Hyperparameter tuning was applied on three machine learning-based method base classifier and the meta learner which was performed using GridSearchCV. GridSearchCV perform exhaustive search over the specific prefix parameter grid to find the best hyperparameter for each classifier. The stacking ensemble is then applied to further enhance the predictive performance. Stacking involves in ensemble the prediction from multiple base classifiers into a meta classifier to make final prediction. The meta learner adopted in this experiment was CatBoost classifier. Stacking approach leverages the diverse strengths from each base classifier to improve the overall performance.

Figure 4.6 shows the confusion matrix of ensemble model. The model precisely predicted that 4550 customers will not churn, and 1633 customer will churn. The model predicts wrongly on 2750 not churn customer to churn, and 1267 churn customer to not churn, which is lesser than the correct prediction. Additionally, Figure 4.7 shows the ROC curve of ensemble model.

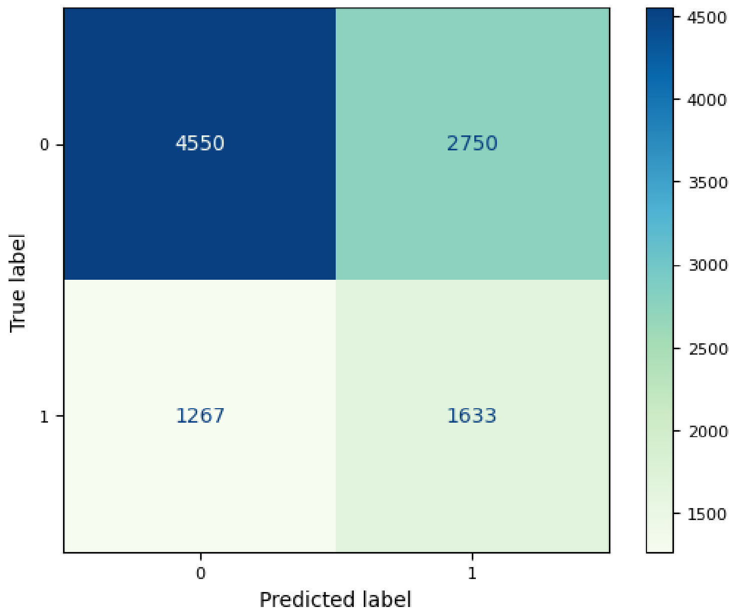

Figure 4.6. Confusion Matrix of Ensemble Model

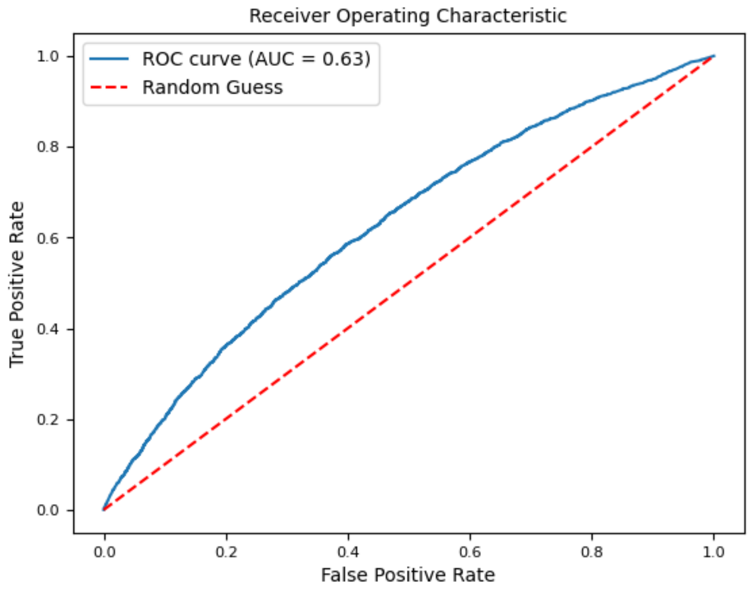

Figure 4.7. ROC Curve of Ensemble Model

Figure 4.8 shows the accuracy, precision, recall, f1-score, and AUC. The proposed ensemble model achieved the accuracy of 60.62%. Indeed, the proposed ensemble model performed well on precision and recall. It achieved a precision of 66.57% which indicate that when it had 66.57% of correctness on predicted a positive instance and minimize the false positives rate. Additionally, the model obtained 60.62% of recall. The model can capture considerable part of positive instances. The combination of high precision and recall led to a higher F1 score of 62.40%. The higher F1 score indicates a good balance between the ability to correctly classify positive instances and able to capture a significant portion of positive instances. The AUC represents the discriminate ability between positive and negative instances. A higher AUC indicates better discriminative power which at 63%.

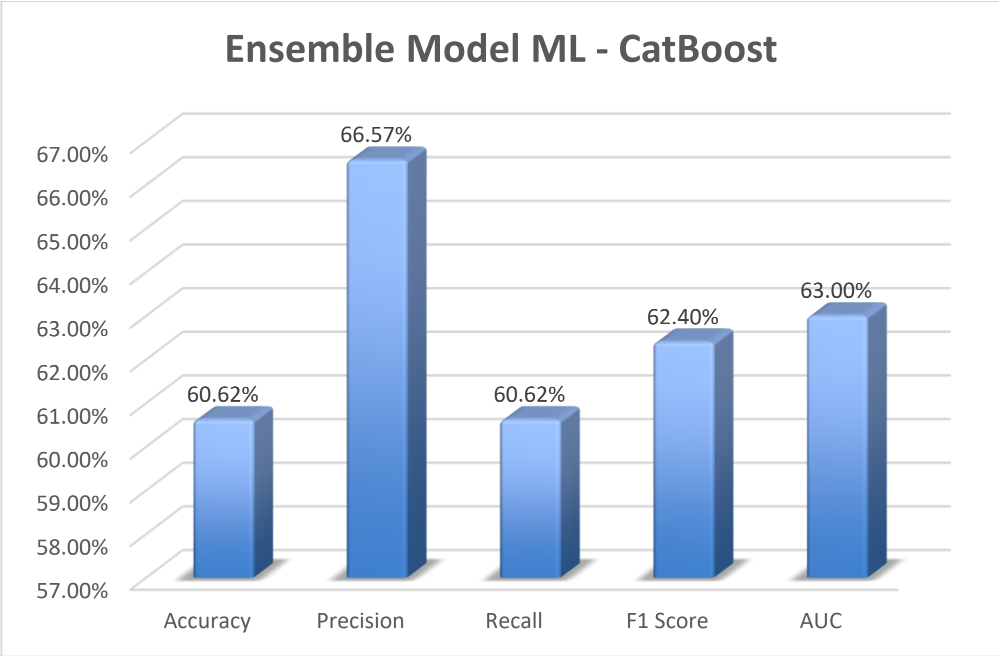

Figure 4.8. Performance Metric Ensemble Model

CHAPTER 5

Discussion and Conclusion

5.1 Comparison with Other Models

A performance comparison between AdaBoost, XGBoost, Deep Neural Network, TL-DeepE, and proposed ensemble model is shown in Table 5.1. The evaluation is based on various performance metrics such as accuracy, precision, recall, F1 score, and AUC. The proposed ensemble achieved the highest scores on precision, recall, and f1 score compared to the other models. The higher precision which indicates that the model had better ability on identify the churn which is particular crucial in churn prediction, as it identify the customer who likely to chur. A Higher recall indicate that model had better ability in identifying the true positive instances, while a better f1-score reflects the model had better balance between the precision and recall. In terms of accuracy and AUC, the proposed model outperforms Logistic Regression and AdaBoost. This shows that the suggested ensemble model works well in terms of the true positive rate and false positive rate and has strong discriminative power. However, the AUC of proposed ensemble model is slightly lower than the Deep Neural Network and XGBoost. Although the AUC of proposed ensemble model is slightly lower, it is crucial to take into account on the over performance measures. In comparison to Deep Neural Network, the ensemble model shows higher precision and recall which indicate the ability to identify churn.

Table 5.1: Performance Comparison between proposed ensemble model and other classifiers

Classifier	Accuracy	Precision	Recall	F1 Score	AUC
Logistic Regression (Wu et al., 2021)	0.576	0.3516	0.5367	0.4109	0.5866
AdaBoost (Wu et al., 2021)	0.5863	0.3653	0.4932	0.4052	0.5723
XGBoost (Nguyen et al., 2021)	-	-	-	-	0.6618

Deep Neural Network (Albrecht et al., 2020)	-	0.401	0.484	-	0.645
TL-DeepE (Ahmed et al., 2019)	0.682	-	-	-	-
Proposed Ensemble Model	0.6062	0.6657	0.6062	0.624	0.63

5.2 Conclusion

Telecom industry has been highly competitive in recent years. This study aims to analyse the CNN on customer churn prediction. Aforementioned, the performance on built CNN standalone is not that optimal. Therefore, the CNN is proposed to ensemble with other machine learning models. This study proposes a telecom-based customer churn prediction model on the stacking ensemble of machine learning and deep learning which consist of LR, DT, SVM, and CNN. The proposed ensemble model focuses on the Cell2cell dataset.

This paper was beginning with discussion on churn prediction. Chapter 1 introduces the overview of churn prediction on telecom industry. Not only that, Chapter 1 also included the problem statement, project scope, and objectives and project management, and project organization. In Chapter 2, the literature review related to machine learning, ensemble learning and deep learning on churn prediction was conducted.

Since most of the machine learning methods cannot deal with the raw unprocessed data, data pre-processing and Exploratory Data (EDA) is adopted to improve the quality of the data from raw data and gain insight and pattern from the data. Moreover, the SMOTE technique is applied to deal with the imbalance nature of the selected dataset. The pre-processing steps and resampling were included in Chapter 3. Not only that, Chapter 3 also included the description of the adopted model. The experiment set up and analysis is presented in Chapter 4.

In conclusion, the proposed model results show that the model achieves an accuracy of 60.62%, precision of 66.57%, recall of 60.62%, f1-score of 62.40%, and AUC of 63%. These performance metrics indicate the effectiveness of the stacking ensemble model in predicting customer churn.

5.3 Suggestion for Future Work

The studies of CNN further improving on customer churn prediction can be further improving. In order to further enhance the performance of CNN model and studies on its potential, future work can be conducted by integrating transfer learning techniques into the CNN model to predict churn. The area of transfer learning on CNN model have been used to research the churn prediction on Cell2cell dataset (Ahmed et al., 2019). Transfer Learning with CNN model involves in using pre-trained CNN model that trained on large datasets to improve the performance of predictive. Transfer Learning showed the improvement to the predictive on churn prediction.

REFERENCES

Abbasimehr, H., Toosi, K. N., & Tarokh, M. J. (2011). A Neuro-Fuzzy Classifier for Customer Churn Prediction Mostafa Setak. *International Journal of Computer Applications*, *19*(8), 975–8887. http://www.ics.uci.edu/

Ahmed, U., Khan, A., Khan, S. H., Basit, A., Haq, I. U., & Lee, Y. S. (2019). *Transfer Learning and Meta Classification Based Deep Churn Prediction System for Telecom Industry*. https://arxiv.org/abs/1901.06091v2

Albrecht, T., Baier, D., & Bayer, D. (2020). *Churn Analysis Using Deep Learning: Customer Classification from a Practical Point of View*. https://doi.org/10.5445/KSP/1000098012/04

Almufadi, N., Mustafa Qamar, A., & Author, C. (2022). *Deep Convolutional Neural Network Based Churn Prediction for Telecommunication Industry*. https://doi.org/10.32604/csse.2022.025029

Ben Jabeur, S., Gharib, C., Mefteh-Wali, S., & Arfi, W. Ben. (2021). CatBoost model and artificial intelligence techniques for corporate failure prediction. *Technological Forecasting & Social Change*, *166*, 120658. https://doi.org/10.1016/j.techfore.2021.120658

De Bock, K. W., & Poel, D. Van Den. (2011). An empirical evaluation of rotation-based ensemble classifiers for customer churn prediction. *Expert Systems with Applications*, *38*(10), 12293–12301. https://doi.org/10.1016/j.eswa.2011.04.007

Deng, Y., Li, D., Yang, L., Tang, J., & Zhao, J. (2021). Analysis and prediction of bank user churn based on ensemble learning algorithm. *Proceedings of 2021 IEEE International Conference on Power Electronics, Computer Applications, ICPECA 2021*, 288–291. https://doi.org/10.1109/ICPECA51329.2021.9362520

Facts and Figures 2022 - Subscriptions. Retrieved May 29, 2023, from https://www.itu.int/itu-d/reports/statistics/2022/11/24/ff22-subscriptions/

From Autoencoder to Beta-VAE | Lil'Log. Retrieved June 6, 2023, from https://lilianweng.github.io/posts/2018-08-12-vae/

Hammoudeh, A., Fraihat, M., & Almomani, M. (2019). Selective ensemble model for telecom churn prediction. *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology, JEEIT 2019 - Proceedings,* 485–487. https://doi.org/10.1109/JEEIT.2019.8717406

Idris, A., Rizwan, M., & Khan, A. (2012). Churn prediction in telecom using Random Forest and PSO based data balancing in combination with various feature selection strategies. *Computers & Electrical Engineering*, *38*(6), 1808–1819. https://doi.org/10.1016/J.COMPELECENG.2012.09.001

Karanovic, M., Popovac, M., Sladojevic, S., Arsenovic, M., & Stefanovic, D. (2018). Telecommunication Services Churn Prediction - Deep Learning Approach. *2018 26th Telecommunications Forum, TELFOR 2018 - Proceedings*. https://doi.org/10.1109/TELFOR.2018.8612067

Kingma, D. P., & Welling, M. (2013). *Auto-Encoding Variational Bayes*.

Liu, R., Ali, S., Bilal, S. F., Sakhawat, Z., Imran, A., Almuhaimeed, A., Alzahrani, A., & Sun, G. (2022). An Intelligent Hybrid Scheme for Customer Churn Prediction Integrating Clustering and Classification Algorithms. *Applied Sciences (Switzerland)*, *12*(18). https://doi.org/10.3390/APP12189355

Mohammad, N. I., Ismail, S. A., Kama, M. N., Yusop, O. M., & Azmi, A. (2019, August 26). Customer Churn Prediction in Telecommunication Industry Using Machine Learning Classifiers. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3387168.3387219

Nguyen, N. N., & Duong, A. T. (2021). *Comparison of Two Main Approaches for Handling Imbalanced Data in Churn Prediction Problem*. https://doi.org/10.12720/jait.12.1.29-35

Sill, J., Takacs, G., Mackey, L., & Lin, D. (2009). *Feature-Weighted Linear Stacking*. http://www.netflixprize.com

Tariq, M. U., Babar, M., Poulin, M., & Khattak, A. S. (2022). Distributed model for customer churn prediction using convolutional neural network. *Journal of Modelling in Management*, *17*(3), 853–863. https://doi.org/10.1108/JM2-01-2021-0032

telecom churn (cell2cell) | Kaggle. Retrieved May 29, 2023, from https://www.kaggle.com/datasets/jpacse/datasets-for-churn-telecom

Usman, M., Ahmad, W., & Fong, A. (2021). Design and Implementation of a System for Comparative Analysis of Learning Architectures for Churn Prediction. *IEEE Communications Magazine*, *59*(9), 86–90. https://doi.org/10.1109/MCOM.110.2100145

Wael Fujo, S., Subramanian, S., Ahmad Khder, M., Fujo, W., & Khder, A. (2022). Customer Churn Prediction in Telecommunication Industry Using Deep Learning. *Information Sciences Letters*, *11*(1). https://doi.org/10.18576/isl/110120

Wang, X., Nguyen, K., Vietnam Hanoi, I., Binh Nguyen, V. P., & Nguyen, B. P. (2020). *Churn Prediction using Ensemble Learning*. https://doi.org/10.1145/3380688.3380710

Wu, S., Yau, W. C., Ong, T. S., & Chong, S. C. (2021). Integrated Churn Prediction and Customer Segmentation Framework for Telco Business. *IEEE Access*, *9*, 62118–62136. https://doi.org/10.1109/ACCESS.2021.3073776

Zilvan, V., Ramdan, A., Heryana, A., Krisnandi, D., Suryawati, E., Yuwana, R. S., Kusumo, R. B. S., & Pardede, H. F. (2022). Convolutional variational autoencoder-based feature learning for automatic tea clone recognition. *Journal of King Saud University - Computer and Information Sciences*, *34*(6), 3332–3342. https://doi.org/10.1016/j.jksuci.2021.01.020

APPENDICES

Appendix A: Final Year Project Meeting Log

MULTIMEDIA UNIVERSITY

Faculty of Information Science and Technology (FIST)

Final Year Project Meeting Log

MEETING DATE: 29/3/2023	**MEETING NO.:** /
PROJECT ID: T812293	
PROJECT TITLE : Analysis of Convolutional Neural Networks on customer churn prediction	
SESSION : 2022/2023	**SUPERVISOR :** Dr. Pang Ying Han
STUDENT ID & Name: 1191202488 Tan Yan Lin	**CO- SUPERVISOR :**

All to be filled in by student

1. WORK DONE [Please write the details of the work done after the last meeting.]

- research on churn prediction.

2. WORK TO BE DONE

understand the entire process

3. PROBLEMS ENCOUNTERED

-

4. COMMENTS

DR. PANG YING HAN
Associate Professor
Faculty of Information Science and Technology
Multimedia University
Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia

Supervisor's Signature & Stamp

Co-Supervisor's Signature & Stamp (if any)

Student's Signature

NOTES:

1. Items 1 – 3 are to be completed by the students before coming for the meeting. Item 4 is to be completed by the supervisor.
2. For FYP Phase 1, total six log sheets are to be submitted (every other week*).
3. For FYP Phase 2, total six log sheets are to be submitted (every other week**).
4. Log sheets are compulsory assessment criteria for FYP. Student who fails to meet the requirements of log sheets will not be allowed to submit FYP report.

*: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the first trimester (week 11: report submission, weeks 13 & 14: presentation)

**: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the second trimester (week 11: report submission, weeks 13 & 14: presentation)

MULTIMEDIA UNIVERSITY

Faculty of Information Science and Technology (FIST)
Final Year Project Meeting Log

MEETING DATE: 3/4/2023	**MEETING NO.:** 7
PROJECT ID: T812293	
PROJECT TITLE : Analysis of Convolutional Neural Networks on customer churn prediction	
SESSION : 2022/2023	**SUPERVISOR :** Dr. Pang Ying Han
STUDENT ID & Name: 1191202483 Tan Yan Lin	**CO- SUPERVISOR :**

All to be filled in by student

1. WORK DONE [Please write the details of the work done after the last meeting.]

Tried different CCN

2. WORK TO BE DONE

try diff conv. based models from traditional CNN/TCN to modified eg:

i/p → filter size A model 1 →

i/p → model 2 →

filter size B.

3. PROBLEMS ENCOUNTERED

-

4. COMMENTS

1Nth6 F1 95%

DR. PANG YING HAN
Associate Professor
Faculty of Information Science and Technology
Multimedia University
Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia

Supervisor's Signature & Stamp	Co-Supervisor's Signature & Stamp (if any)	Student's Signature

NOTES:

1. Items 1 – 3 are to be completed by the students before coming for the meeting. Item 4 is to be completed by the supervisor.
2. For FYP Phase 1, total six log sheets are to be submitted (every other week*).
3. For FYP Phase 2, total six log sheets are to be submitted (every other week**).
4. Log sheets are compulsory assessment criteria for FYP. Student who fails to meet the requirements of log sheets will not be allowed to submit FYP report.

*: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the first trimester (week 11: report submission, weeks 13 & 14: presentation)

**: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the second trimester (week 11: report submission, weeks 13 & 14: presentation)

MULTIMEDIA UNIVERSITY

Faculty of Information Science and Technology (FIST)
Final Year Project Meeting Log

MEETING DATE: 10/4/2023	**MEETING NO.:** 3
PROJECT ID: T812293	
PROJECT TITLE : Analysis of Convolutional Neural Networks On Customer Churn Prediction	
SESSION : 2022/2023	**SUPERVISOR :** Dr. Pang Ying Han
STUDENT ID & Name: 1191202488 Tan Yan Lin	**CO- SUPERVISOR :**

All to be filled in by student

1. WORK DONE [Please write the details of the work done after the last meeting.] Tried on ~~a~~ different CNN Imgd. CNN (~~transfor~~ (encoder).
2. WORK TO BE DONE direction : structured data → image → process
3. PROBLEMS ENCOUNTERED - ① spectrum (technique) ② value ③
4. COMMENTS

DR. PANG YING HAN
Associate Professor
Faculty of Information Science and Technology
Multimedia University
Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia

Supervisor's Signature & Stamp | Co-Supervisor's Signature & Stamp (if any) | Student's Signature

NOTES:

1. Items 1 – 3 are to be completed by the students before coming for the meeting. Item 4 is to be completed by the supervisor.
2. For FYP Phase 1, total six log sheets are to be submitted (every other week*).
3. For FYP Phase 2, total six log sheets are to be submitted (every other week**).
4. Log sheets are compulsory assessment criteria for FYP. Student who fails to meet the requirements of log sheets will not be allowed to submit FYP report.

*: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the first trimester (week 11: report submission, weeks 13 & 14: presentation)
**: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the second trimester (week 11: report submission, weeks 13 & 14: presentation)

MULTIMEDIA UNIVERSITY

Faculty of Information Science and Technology (FIST)
Final Year Project Meeting Log

MEETING DATE: 26/4/2023	**MEETING NO.:** 4
PROJECT ID: T8l2293	
PROJECT TITLE : Analysis of Convolutional Neural Network on customer churn prediction	
SESSION : 2022/2023	**SUPERVISOR :** Dr. Pang Ying Han.
STUDENT ID & Name: 1191202488 Tan Yan Lin	**CO-SUPERVISOR :**

All to be filled in by student

1. WORK DONE [Please write the details of the work done after the last meeting.]

- Tried CNN with VAE

2. WORK TO BE DONE

Try ensemble CNN with machine Learning model

3. PROBLEMS ENCOUNTERED

-

4. COMMENTS

PANG YING HAN
Associate Professor
Faculty of Information Science and Technology
Multimedia University
Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia

Supervisor's Signature & Stamp

Co-Supervisor's Signature & Stamp (if any)

Student's Signature

NOTES:

1. Items 1 – 3 are to be completed by the students before coming for the meeting. Item 4 is to be completed by the supervisor.
2. For FYP Phase 1, total six log sheets are to be submitted (every other week*).
3. For FYP Phase 2, total six log sheets are to be submitted (every other week**).
4. Log sheets are compulsory assessment criteria for FYP. Student who fails to meet the requirements of log sheets will not be allowed to submit FYP report.

*: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the first trimester (week 11: report submission, weeks 13 & 14: presentation)
**: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the second trimester (week 11: report submission, weeks 13 & 14: presentation)

MULTIMEDIA UNIVERSITY

Faculty of Information Science and Technology (FIST)

Final Year Project Meeting Log

MEETING DATE: 11/5/2023	**MEETING NO.:** 5
PROJECT ID: TB12293	
PROJECT TITLE : Analysis of Convolutional Neural Network on customer churn prediction.	
SESSION : 2022/2023	**SUPERVISOR :** Dr. Pang Ying Han
STUDENT ID & Name: 1191202468 Tan Yan Lin	**CO- SUPERVISOR :**

All to be filled in by student

1. WORK DONE [Please write the details of the work done after the last meeting.]

- Try on ensemble model

2. WORK TO BE DONE

prepare journal writing ⇒ imt FYP1 report.

3. PROBLEMS ENCOUNTERED

4. COMMENTS

DR. PANG YING HAN
Associate Professor
Faculty of Information Science and Technology
Multimedia University
Melaka, Malaysia

Supervisor's Signature & Stamp	Co-Supervisor's Signature & Stamp (if any)	Student's Signature

NOTES:

1. Items 1 – 3 are to be completed by the students before coming for the meeting. Item 4 is to be completed by the supervisor.
2. For FYP Phase 1, total six log sheets are to be submitted (every other week*).
3. For FYP Phase 2, total six log sheets are to be submitted (every other week**).
4. Log sheets are compulsory assessment criteria for FYP. Student who fails to meet the requirements of log sheets will not be allowed to submit FYP report.

*: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the first trimester (week 11: report submission, weeks 13 & 14: presentation)

**: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the second trimester (week 11: report submission, weeks 13 & 14: presentation)

MULTIMEDIA UNIVERSITY

Faculty of Information Science and Technology (FIST)
Final Year Project Meeting Log

MEETING DATE: 24 / 5 / 2023	MEETING NO.: 6
PROJECT ID: T8I 22 93	
PROJECT TITLE : Analysis of Convolutional Neural Network on customer churn prediction	
SESSION : 2022 / 2023	SUPERVISOR : Dr. Pang Ying Han.
STUDENT ID & Name: 1191202488 Tan Yan Lin	CO-SUPERVISOR :

All to be filled in by student

1. WORK DONE [Please write the details of the work done after the last meeting.] - journal writing
2. WORK TO BE DONE - comprehensive writing - format, illustrate
3. PROBLEMS ENCOUNTERED - check format (ref, caption). -
4. COMMENTS DR. PANG YING HAN Associate Professor Faculty of Information Science and Technology Multimedia University Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia

Supervisor's Signature & Stamp	Co-Supervisor's Signature & Stamp (if any)	Student's Signature

NOTES:
1. Items 1 – 3 are to be completed by the students before coming for the meeting. Item 4 is to be completed by the supervisor.
2. For FYP Phase 1, total six log sheets are to be submitted (every other week*).
3. For FYP Phase 2, total six log sheets are to be submitted (every other week**).
4. Log sheets are compulsory assessment criteria for FYP. Student who fails to meet the requirements of log sheets will not be allowed to submit FYP report.

*: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the first trimester (week 11: report submission, weeks 13 & 14: presentation)
**: week 1, 3, 5, 7, 9, 11 or 2, 4, 6, 8, 10 of the second trimester (week 11: report submission, weeks 13 & 14: presentation)

FYP for FIST: Attendance List
Session 2: 13/04/2023

The following students have attended the FYP session (FIST) with the library.

Trainer:

Nurul Irtika Mohamad Nori
Librarian, Siti Hasmah Digital library

No.	Full Name	MMU ID No.	FYP title
1	Hau Khai Ming	1201302885	NFT application
2	Nicholas Lee Ze Xuan	1181102808	2.5 image extractor
3	Gee Wen Guan	1191201095	Pet Adoption System
4	Dwyahsree A/P Bhaskaran	1191201552	Develop A Consumer Purchasing Propensity Model
5	Foo Chuan Ping	1181203180	eBook website
6	BEH JI HUI	1191200804	Weather Reporting System
7	IRFAN HAFIZ BIN ZULKIFLI	1191102148	AI-BASED ANGER DETECTION
8	Wong Chung Vy	1181203221	Fire and Smoke Detection by Computer Vision
9	Tan Tian Yao	1181103388	NFT Application 3
10	WONG JING WEI	1181200535	AI-Based Expense Management App
11	NG CHEE CHIEN	1191200922	NFC SECURITY
12	HOO SHI YAO	1191200506	Pc desktop control using gui
13	Raisa Raad Khan	1191202087	IoT based trash separation system
14	CAHIM YI FU	1191202682	Its my way
15	Liyu Guang Jie	1191200921	ONLINE CLOTHING SHOPPING MANAGEMENT SYSTEM
16	Muhammad Faris Bin Hanafiah	1201302744	Cloud Based Milk Siblings System
17	NG HUI PIN	1191201604	JOB SEARCH WEB
18	Lan Si Ying	1201300254	Personalized AR-based mobile app for fashion technology
19	Ataul Haque Ove	1191202480	IoT based Rotten Food Detector
20	Rishan Muhammed	1191302438	Prediction of Stock Market using deep learning
21	Umar Abdul Aziz Bin Shahraan	1171200607	Library Management System
22	GAN YUN SIANG	1191200221	Smart Gardening System
23	Kor Jia Hui	1191200550	SCHOOLS
24	MICHAEL ANTHONY VARGHESE	1191101225	Airline Reservation System
25	Gan Kai Boon	1181103405	QUIZ SYSTEM USING NATURAL USER INTERFACE

26	ALVIN CHOO CHUNG HOW	118102681	UNIVERSITY VENUE MANAGEMENT
27	TAN KAI CHONG	1161101298	Customer Market Basket Analysis Using Apriori And Fpgrowth Algorithms
28	FAEZ	1171201452	TRAFFIC ACCIDENT EVENTS DETECTION FROM SOCIAL MEDIA PLATFORM
29	ANIS SYAFIQAH BINTI JUSOH	1191102183	CRISIS COMMUNICATION DURING FLOOD DISASTER
30	POO ZHIBIN	1191301874	time series forecasting stocks analysis
31	Nurhakim B. Mohamed Jasmin	1171202252	IoT Based Home Intrusion Detection System
32	AMIRAH ZULAIKHA BINTI RAMLY	1201302228	BLOCKCHAIN-BASED ELECTRONIC HEALTH RECORDS MANAGEMENT SYSTEM
33	ANISH A/L R GOPAL	1141128017	Recommendation system for personalized nutrition plans
34	LIN HUAN LONG	1191200243	Smart Alarm Clock System
35	Ley Chin Chong	1191202597	UNIVERSITY
36	NG KOK HANG	1191202169	GESTURE RECOGNITION FOR VEHICLE INFOTAINMENT SYSTEM
37	Sashwin Naidu A/L Maran	1281202048	Job Searching For Internship System
38	Al ghamdi omar saeed o	1191302763	SECURING HEALTHCARE DATA WITH BLOCKCHAIN TECHNOLOGY
39	CHEW JIA HAO	1171201061	ONLINE INVENTORY SYSTEM FOR FACULTY AND STUDENTS
40	Haress A/L Sivakumar	1181202537	Anti Keylogger System
41	Amodi	1201302204	FACE RECOGNITION WITH 2.5D FOR STAFF ATTENDANCE
42	Chai Chin Heng	1191202373	Facial features for drunk driving driver
43	AMNI BAZILAH HUSNA BINTI NAZARUDIN	1191102157	Penetration Testing Approach for IOT Device
44	Marcus Ang Khai Kiat	1181203412	Identification of weaknesses of chatgpt in drone
45	Dominic Lee Guang Zhen	1191202352	An Income-level Classification Model Based On Demographic And Economic Data
46	LIM LE XIANG	1191200843	Hightway traffic analysis using computer vision technology
47	KONG KEIN WAH	1191201503	SMART RENTAL SYSTEM WITH FACIAL RECOGNITION
48	Ng Tze Yang	1191202079	Performance analysis of security attack on vanet
49	HEYMANATHAN A/L VELUMURUGAN	1181201935	WEB-BASED MOTORCYCLE RETAILING SYSTEM
50	CHONG YEW SAM	1191201277	Pet e-tag management
51	Eric Cheah Jian Peng	1191201198	BUILD AN AI CHATBOT FOR E-COMMERCE USING THE CHATGPT API
52	TAN YAN LIN	1191202488	Analysis of Convolutional Neural Network on customer churn prediction
53	Gan Ghim Hong	1181203098	REAL-TIME IOT-BASED AIR QUALITY MONITORING AND ANALYSIS
54	NAZATUL NIESYA BINTI SHAHAROM	1201302629	MEDICAL SUPPLY CHAIN USING BLOCKCHAIN TECHNOLOGY
55	Loo Wei Jun	1171103833	MUTUAL FUNDS VISUALISER AND ADVISOR TOOL
56	Yen Han Wei	1181203089	IOT-ENABLED LABORATORY MONITORING AND CONTROL
57	Ryan Wee Mo Xian	1191102483	TRAFFIC SURVEILLANCE FOR ROAD SAFETY (BACKEND)
58	Loo Chuan Wei	1201301033	Daycare pickup system

Appendix B: Checklist for FYP Interim Submission

Faculty of Information Science and Technology (FIST)

Checklist for Interim Report Submission
(To be filled in by Student)

STUDENT'S DETAILS

Project Code	**T812293**
Name	**TAN YAN LIN**
ID No	**1191202488**
Title of Thesis	**ANALYSIS OF CONVOLUTIONAL NEURAL NETWORKS ON CUSTOMER CHURN PREDICTION**
Supervisor Name	**Dr. Pang Ying Han**

REPORT ARRANGEMENT	√	**Comments** (if any differences)
1. Cover of The Interim Report	√	
2. Title Page of the Interim Report	√	
3 Copyright page of I Interim Report	√	
4. Declaration Page of Interim report	√	
5. Acknowledgement	√	
6. Table of Contents	√	
7. Abstract	√	
8. List of Tables	√	
9. List of Figures	√	
10. List of Symbols	√	
11. List of Appendices	√	
12. Chapter 1: Introduction	√	
13. Chapter 2: Literature Review	√	
14. Chapter 3: Methodology	√	
15. Chapter 4: Experimental and Analysis	√	
16. Chapter 5: Discussion and Conclusion	√	
17. References – APA style	√	
18. Appendices	√	
19. CD/ DVD and envelope as shown in Appendix K	N/A	
20. Attachment : FYP Meeting Logs (all) 1 set	√	

FORMAT OF REPORT	√	**Comments**
1. Page Numbering	√	
2. Font and Type Face	√	
3. Font Cover	√	
4. Tables and Figures	√	
5. Comb Bind	N/A	
6. Colour of the Front Cover	N/A	
7. Number of words > 5000 (Main content only)	√	

Checked by,

(7/6/2023)

Student's Signature & Date