IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/45AI/2025/4526625/
Vol. 2025 (20), Article ID 4526625

Language Model-Enhanced Feature Engineering Framework for Customer Churn Analysis*

Maryam SHAHABIKARGAR, Amin BEHESHTI, Saleh AFZOON,
Jin FOO, Xuyun ZHANG and Nasrin SHABANI

School of Computing, Macquarie University, Sydney, Australia

Correspondence should be addressed to: Maryam SHAHABIKARGAR, maryam.shahabi-kargar@hdr.mq.edu.au

* Presented at the 45th IBIMA International Conference, 25-26 June 2025, Cordoba, Spain

Abstract

Customer churn remains a major concern across industries, especially with the rising importance of customer retention over acquisition. While prior research has focused heavily on structured data, the potential of customer-generated textual data, such as chat logs and feedback, remains underutilized. This study addresses this gap by introducing a novel feature engineering framework that leverages Language Models (LMs), to extract meaningful insights from unstructured text data for churn prediction. We propose a multi-stage pipeline that combines domain expertise with LM capabilities to generate interaction features, sentiment labels, emotional tone scores, and topic- based features from customer chat data. Additionally, we introduce a new composite metric, the Normalized-weighted Churn Score, which integrates expert-assigned topic weights with language model outputs. The framework was evaluated using a churn dataset containing structured and unstructured data. Results show that incorporating LM-enhanced features significantly boosts model performance across multiple classifiers. Notably, models using our enriched feature outperformed traditional baselines, achieving an F1-score increase of over 26%. The findings emphasize the critical role of text analytics and hybrid feature engineering in advancing churn pre- diction and offer a scalable approach for integrating domain knowledge with modern NLP techniques.

Keywords: Customer Churn Analysis, Feature Engineering, Language Models, Textual Data Processing

Introduction

Customers are one of the main sources of profit for businesses. A customer's tendency to stop doing business with a company within a given period or contract is known as customer churn (Tueanrat et al., 2021). Attracting a new customer costs 5–10 times more than retaining an existing one (Wu et al., 2022). On the other hand, retaining customers is challenging due to complex customer behavior and diverse influencing factors. Most businesses experience customer churn and struggle to predict when customers will leave (Knowles, 2021). Therefore, it is crucial to prevent customer churn by using enhanced predictive analysis as well as Machine Learning (ML) workflows.

ML workflows rely on the interplay of data, features, and models to generate insights, with the feature engineering process and model choice significantly impacting one another (Brooks et al., 2015). Effective feature engineering

Cite this Article as: Maryam SHAHABIKARGAR, Amin BEHESHTI, Saleh AFZOON, Jin FOO, Xuyun ZHANG and Nasrin SHABANI, Vol. 2025 (20) "Language Model-Enhanced Feature Engineering Framework for Customer Churn Analysis " Communications of International Proceedings, Vol. 2025 (20), Article ID 4526625, https://doi.org/10.5171/2025.4526625

simplifies modelling and enhances performance, whereas poor features may necessitate more complex models to achieve similar outcomes (Zheng and Casari, 2018). Previous churn studies have primarily focused on utilizing numerical and categorical features. However, customer-generated textual data, such as chat logs, and the derivative features extracted from it contain valuable insights into customers' cognitive status (Ma and Zhang, 2019) and propensity to churn or stay (Vo et al., 2021). Despite its potential, this type of data is often overlooked in existing research (De and Prabu, 2022).

Language Models (LMs), particularly those based on transformer architectures like BERT (Kenton and Toutanova, 2019) and GPT (Brown et al., 2020), have transformed Natural Language Processing (NLP) by enabling machines to generate and interpret human-like text. These models, when fine-tuned for specific tasks such as Sentiment Analysis (SA) (Sahoo et al., 2023), Emotional Tone Detection (ETD) (Hartmann, 2022), and Topic Modelling (TM) (Liu, 2019), allow for precise extraction of valuable insights from textual data. SA identifies the polarity of text (positive, negative, or neutral), while ETD provides a deeper understanding of emotions, such as joy, anger, or sadness (Mohammad and Turney, 2013). TM, often performed using models like Latent Dirichlet Allocation or transformer-based methods, organizes text into thematic clusters, revealing the underlying subjects of discourse (Blei et al., 2003).

In customer churn analysis, fine-tuned LMs enhance feature engineering by extracting meaningful signals from unstructured text, such as customer reviews, complaints, or social media feedback. Sentiment and emotional tone analysis help identify dissatisfied customers or escalating frustration, both of which are strong predictors of churn (Abou el Kassem et al., 2020). Similarly, TM highlights recurring complaints or preferences, enabling companies to address specific customer needs. These enriched textual features, when combined with structured data like transaction history or demographic information, improve the performance of classifiers in churn prediction tasks, offering a comprehensive understanding of customer behavior and retention factors (Verbeke et al., 2012).

This study introduces an LM-based feature engineering framework for churn analysis, integrating LMs and domain expertise to generate meaningful features and enhance the performance of ML algorithms. Fine-tuned LMs were used to analyze textual data for SA, ETD, and TM. Expert-guided topics and their assigned weights were leveraged to create a novel feature, the Normalized-weighted Churn Score (NWCS), which combines normalized topic scores with expert-assigned weights. This framework highlights the importance of textual data, domain expertise, and LMs in advancing feature engineering and understanding complex customer behavior.

Background

Customer Experience and Churn

Customer experience, being a dynamic process, is conceptualized as a customer's journey with a firm over time during the purchase cycle across multiple touchpoints (Lemon and Verhoef, 2016). The term touchpoint refers to service or product interactions that a customer engages with, either directly or indirectly, related to a specific brand, ultimately influencing their views and assessments of the brand in general and the customer journey in particular (Tueanrat et al., 2021). Therefore, the customer journey, defined as the entirety of a customer's interactions with a company (Seobility Wiki, 2021), plays a vital role in determining satisfaction with the product or service and could ultimately impact customer success and churn.

Customer churn is commonly used to describe a customer's tendency to stop doing business with a company during a given time period or contract (Tueanrat et al., 2021). Many businesses struggle with customer churn because they lack the tools to deliver competitive customer experience. Without a clear picture of their customers, organizations face loss of revenue, poor visibility into customer data, and increased customer relationship crises, including an inability to predict when customers may churn (Knowles, 2021).

Customer Churn Data and Features

Customer data typically includes a range of valuable information that provides insights into the dynamics of the customer-company relationship. The majority of previous churn studies have focused on structured data (Alboukaey et al., 2020; Höppner et al., 2020), while a smaller body of work has incorporated both structured and unstructured data (Vo et al., 2021; Slof et al., 2021). Structured or numerical data refers to organized, tabular information primarily consisting of quantitative values (Mitrović et al., 2018). A key subset of structured data is demographic features, attributes such as a customer's age, income, and gender. Another important category is product-usage features, which capture how customers interact with a product or service, including usage duration, accessed functionalities, and RFM (Recency, Frequency, Monetary)-based metrics. Research has demonstrated that recency and frequency of customer engagement are particularly valuable for predicting churn (Tamaddoni et al., 2016). For instance, the number of outgoing calls a customer makes daily is an example of a product-usage feature.

Unstructured data, in contrast, is retained in its original format and generally qualitative in nature. It must be processed by data scientists and engineers before ML algorithms can effectively utilize it. Examples include customer emails, call transcripts, and chat logs. In a study by Slof et al. (2021), topic variables extracted from textual data significantly improved model performance. Similarly, Vo et al. (2021) used text mining and NLP to predict churn, driven by the observation that structured variables often exhibit weak correlation with churn outcomes. Models that incorporated unstructured variables outperformed those that did not in both studies, underlining the importance of information-rich content in consumer-business communication (Slof et al., 2021; Vo et al., 2021).

Support service and satisfaction-related features are also critical in churn analysis. These include customer interactions with support teams, reporting damaged products, complaints about quality, return requests, refunds (Tamaddoni et al., 2016), inquiry volumes, resolution times, and customer feedback or satisfaction ratings (Agrawal et al., 2018; Höppner et al., 2020). Collectively, these dimensions provide a well-rounded view of the customer, helping companies tailor their services and improve satisfaction. Many past studies have emphasized satisfaction and service quality as key factors contributing to churn (Sifa et al., 2014). Despite the value of unstructured textual data, such as chat logs, reviews, and social media posts, its integration into churn prediction frameworks remains limited. Yet, such data offers rich behavioral and emotional insights (Vo et al., 2021; Slof et al., 2021; De and Prabu, 2022).

2.3. *Generative AI in Customer Churn Prediction*

Generative AI has transformed customer churn prediction by enabling the analysis of vast unstructured data using Language Models (LMs). These models excel in processing natural language and provide valuable insights into customer sentiments, behaviors, and churn risk. By leveraging these capabilities, businesses gain deeper understanding and can proactively address churn drivers (Madanchian, 2024).

Fine-tuning LMs for tasks such as Sentiment Analysis (SA), Emotional Tone Detection (ETD), and Topic Modeling (TM) has proven particularly effective. SA uncovers customer emotions and satisfaction levels, while TM identifies themes in feedback, yielding actionable insights. Wang et al. (2018) showed that integrating voice-of-the-customer data and generative models into churn prediction pipelines significantly enhances predictive performance by capturing nuanced emotional cues.

The use of LMs in churn analysis is further supported by recent research. Lghaouch et al. (2024) demonstrated that combining TM and SA enhances the interpretability of customer data. Dias and Antonio (2023) emphasized that models such as XGBoost perform well when applied to industry-specific datasets. Slavchanyk et al. (2024) highlighted the effectiveness of random forest and gradient boosting methods in analyzing historical churn data. Collectively, these studies underscore the potential of generative AI in designing data-driven customer retention strategies.

Method

In this study, we propose a method that integrates generative AI and LMs, advanced ML pipelines, and normalization methods to generate new features to enhance feature engineering for customer churn prediction. The process is divided into five key stages, namely, *A) Input Data Preparation*, *B) Generative AI-Enabled Interaction Feature Generation*, *C)Language Model-based Feature Generation*, *D) Intuitive Feature Generation*, and *E) Enriched Feature-based Classifier Selection*, as shown in Figure1. Stages B, C, and E incorporate a feature selection process to ensure that only reliable and insightful features are used in the corresponding ML models.

Feature selection in this study is conducted using a correlation-based approach to identify and remove highly correlated features to the target variable. After dummy encoding the categorical variables, the correlation between each feature and the target variable, i.e., binary churn, is calculated. Numerical features with a correlation higher than 0.2 or lower than -0.1 and categorical features with a p-value lower than 0.05 would be selected.

To detect multicollinearity, a correlation matrix is computed for the training data, and pairs of features with a correlation greater than 0.7 are flagged and removed. The rationale behind this threshold is to eliminate redundancy in the dataset, as highly correlated features may carry similar information, which can negatively affect model performance. For each pair of highly correlated features, the feature with the lower correlation to the target variable is removed, ensuring that only the most relevant features are retained.

This process helps refine the feature set by retaining features that are most in- formative and reducing the risk of multicollinearity. After removing the identified features, a new correlation matrix is computed for the remaining variables to confirm that the features are less correlated and more independent. This refined feature set is then used for model training, allowing the model to focus on the variables that have the strongest relationship with the target, ultimately improving model accuracy and interpretability.

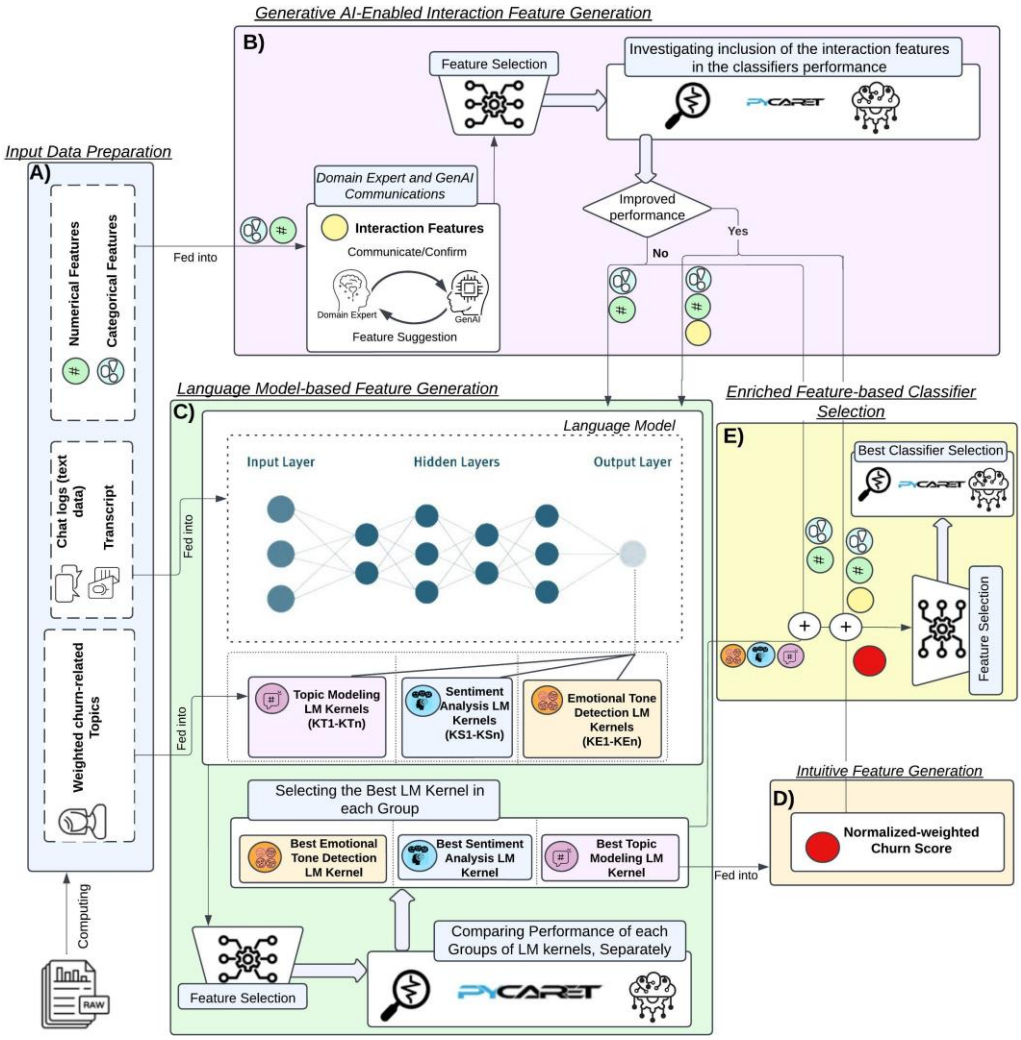

Figure1: Proposed Language Model Enhanced Feature Engineering Framework for Customer Churn Prediction: The framework integrates generative AI, language models, and machine learning pipelines to generate enriched features. It consists of five stages: (A) Input Data Preparation, (B) Generative AI-Enabled Interaction Feature Generation, (C) Language Model-based Feature Generation, (D) Intuitive Feature Generation, and (E) Enriched Feature-based Classifier Selection. The final enriched feature set is fed into a PyCaret[1] pipeline to identify the best-performing churn prediction model.

After the feature selection, all the features are fed into a PyCaret pipeline to compare various ML models and evaluate the impact of the newly generated features on their predictive performance. PyCaret is an open-source, low-code ML library in Python that automates and simplifies the end-to-end ML workflow. It provides a unified interface for tasks such as model training, hyper-parameter tuning, and model evaluation with minimal coding. PyCaret supports a wide range of supervised and unsupervised ML algorithms and integrates seamlessly with popular tools like scikit- learn, XGBoost, and LightGBM.

In this study, the F1-score is used as the evaluation metric for various classification models, as it is particularly suited for churn prediction with imbalanced datasets. The F1-score balances precision and recall, which is crucial for churn scenarios where accurately identifying the minority class (i.e., churners) are essential for cost-effective retention strategies. By focusing on both precision and recall, the F1-score ensures that the model's predictions are reliable and actionable in reducing customer loss. F1-score is calculated as:

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

We also report on the performance of our customer churn prediction models using multiple metrics, in addition to F1-score, including Area Under the Curve (AUC), Precision, and Recall. The AUC measures the model's ability to distinguish between churn and non-churn cases, where higher values indicate better discrimination. Precision measures how many of the customers predicted to churn actually did churn:

$$Precision = \frac{\text{Number of correctly predicted churn customers}}{\text{Total number of customers predicted as churn}}$$

and Recall, measures how well the model identifies all actual churn cases:

$$Recall = \frac{\text{Number of correctly predicted churn customers}}{\text{Total number of actual churn customers}}$$

A brief explanation of the five stages in our methodology is as follows:

Input Data Preparation

The methodology begins with organizing and preparing the input data as a foundational stage. The dataset is assumed to consist of three key feature types, Numerical (e.g., customer tenure, transaction frequency, monetary value), Categorical (e.g., product type, subscription tier), and Text-Based (e.g., chat logs, customer transcripts). In addition to the above-mentioned data type, there is another type of data, i.e., Weighted churn-related topics, suggested by a domain expert, aid in identifying communication patterns associated with churn risk. This diverse feature set enables the model to capture nuanced customer behaviors from structured and unstructured data. Data preparation involved cleaning the dataset by removing rows with null or empty entries, transforming categorical variables using one-hot encoding, label encoding, and dummy encoding, and ensuring the data is structured and ready for analysis.

Generative AI-Enabled Interaction Feature Generation

Interaction features refer to new variables that are created by combining two or more existing features. They allow us to capture the potential relationships between them that may not be obvious when considered individually. This can help reveal hidden patterns in the data that might be critical for tasks such as clustering, classification, or regression. Many ML models, including clustering models, assume that the relationships between variables are linear or independent. However, in real-world data, non-linear relationships often exist between features (Bobbitt, 2022) The interaction features allow us to capture these relationships and improve the performance of the model.

This stage involves collaboration between a domain expert and Generative AI to create meaningful interaction features. The expert first asks Generative AI to explain the dataset and then requests suggestions for interaction features that capture relationships between initial features. After reviewing and validating the suggestions offered, the expert asks Generative AI for formulas to generate the confirmed features. These interaction features, refined through expert validation, are integrated into the feature engineering pipeline to enhance the predictive performance of the churn model.

Language Model-based Feature Generation

In this stage, textual data is processed using LMs to extract additional features that provide insights from unstructured content. Specifically, LMs are employed for SA and ETD, generating corresponding features. Additionally, various LMs are utilized to identify the Conversation Topic of each customer chat log through a Topic Modelling (TM) process. For SA, ETD, and TM features, a set of fine-tuned LMs is compared to evaluate the quality of the generated features. Three groups of LMs are formed, i.e., TM Kernels (KT_1–KT_n), SA Kernels (KS_1–KS_n), and ETD Kernels (KE_1–KE_n). Text data, such as chat logs and transcripts, are separately fed into each kernel within these groups. The kernels in each group are then compared to evaluate how the generated features improve the performance of classification algorithms in

identifying customers' tendency to leave. The best LM kernel in each group is selected based on its individual performance, as determined by the generated feature. Finally, the selected LM kernels from each group are applied together to improve the churn classification task, so that features derived from each of the best kernels will be added to the initial feature set and be used in the final churn classification process.

Intuitive Feature Generation

In this stage, we propose generating an additional feature to incorporate domain- specific insights. This intuitive feature, called Normalized-weighted Churn Score (NWCS), is designed to enhance the feature engineering process and, consequently, improve the predictive performance of customer churn classifiers. The process of generating NWCS includes two steps, i.e., normalization of topic scores and calculation of churn scores.

Given the raw scores $s_1, s_2, \ldots, s_n$ generated by the best topic modelling LM for each topic in a chat log, the normalization step adjusts these scores so that they sum to one. This ensures they are proportional to their relative importance. The normalized score s'_i for topic T_i is computed as:

$$s_i' = \frac{s_i}{\sum_{j=1}^{n} s_j}$$

Where s_i is the raw score for topic T_i, and $\sum_{j=1}^{n} s_j$ is the sum of all raw topic scores for a given chat log. Having s_i divided by the sum of all raw scores in the given chat log ensures that the normalized scores for each chat log sum to one. After normalization, the churn score C is computed as a weighted sum of the normalized topic scores. The normalized scores, s'_i are then weighed by predefined topic weights w_i to calculate the churn score, reflecting the relative importance of each topic in determining the likelihood of churn. The churn score is computed as:

$$C = \sum_{i=1}^{n} w_i . s_i'$$

Where w_i is the predefined weight for topic T_i, and s'_i is the normalized score for topic T_i (calculated in the previous step). With this approach, the churn score integrates both the predicted relevance of each topic (via normalized scores) and the predefined importance of each topic (via the topic weights).

Enriched Feature-based Classifier Selection

The final stage involves selecting the best-performing churn prediction model using features enhanced in earlier stages. This process is automated with PyCaret, a machine learning library for low-code model building and optimization. The enriched feature set, comprising original features, interaction features, language model-generated features, and NWCS feature, undergoes pre-PyCaret feature selection to identify the most impactful features.

On the other hand, Several classifiers are then trained on the selected features, including Decision Tree (DT) (Steinberg, 2009), Gradient Boosting Classifier (GBC) (Friedman, 2001), Extra Trees Classifier (ET) (Geurts et al., 2006), Random Forest (RF) (Breiman, 2001), Extreme Gradient Boosting (XGBoost) (Chen and Guestrin, 2016), and Light Gradient Boosting Machine (LightGBM) (Ke et al., 2017). We used PyCaret's default model comparison pipeline with 10-fold cross-validation. Hyperparameter tuning was conducted using PyCaret's tune model() function, which applies to a randomized grid search strategy optimized for the F1-score. The best- performing model, based on F1-score, was selected for final deployment. This approach ensures robust and efficient model configuration, fully leveraging the enriched feature set.

Experiments

Experimental Setup

The experiments were conducted on a high-performance computational setup featuring an NVIDIA RTX 6000 Ada Generation GPU with 47.99 GB of memory, enabling efficient processing of large datasets and AI-enhanced feature engineering tasks. The experiments utilized the PyTorch framework (version 2.4.1+cu121) with CUDA support, ensuring optimized GPU acceleration for deep learning computations. This setup facilitated efficient, reproducible, and scalable execution of the feature engineering process.

Dataset

For our experiments and to evaluate our approach, we need a churn-related dataset that includes both textual and other data types, such as categorical and numerical. To the best of our knowledge, the only available dataset that meets these criteria is churn-prediction-with-text-and-interpretability[2]. The dataset contains 3,333 rows and 21 columns. Of these, 2,850 samples (85.50%) belong to the 'churn=no' class, while 483 samples (14.50%) belong to the 'churn= yes' class. Each column represents a different feature type. The features include categorical features (e.g., state, international plan, and voice-mail plan) as well as numerical features (e.g., total length of day calls, total charge of evening calls, and total number of night calls). Additionally, the dataset includes a specific column for text data, which contains chat logs between customers and agents.

Implementation

Input Data Preparation

In this stage, data preparation focused on ensuring data quality and consistency. Rows with null or empty chat log entries were removed to retain only meaningful data. Categorical variables were transformed into numerical format using dummy encoding, with redundant categories dropped, and the churn column was encoded using label encoding. These steps collectively enhanced the dataset's structure, ensuring it was clean and suitable for analysis.

Generative AI-Enabled Interaction Feature Generation

In this stage, the communication scenario between the domain expert and the Generative AI are addressed by the following prompts:

- We plan to work on a customer churn problem, using the attached dataset. Explain the dataset and included features.
- What are the possible interaction features that are capable of capturing the relationships between initial features?
- Some of the suggested features, including 'cust-service-day-eve-interaction'[3] etc., sound to have positive effect on the predictive analysis. I would like you to give me the formula corresponding to the creation of these features for further analysis.

Table 1: Comparison of classifiers performance using baseline features and after incorporating interaction features

Model	AUC	Recall	Precision	F1
Baseline features				
GBC	0.8723	0.4970	0.7446	0.5894
XGBoost	0.8504	0.5005	0.6830	0.5752
LightGBM	0.8504	0.4823	0.6711	0.5567
Baseline features + interaction features				

GBC	0.8764	0.5799	0.8135	0.6747
XGBoost	0.8698	0.6168	0.7872	0.6891
LightGBM	0.8705	0.6245	0.7979	0.6980

Confirmed interaction features were generated and added to the initial feature set to compare the performance of classification models before and after their inclusion. An example of an interaction feature is one derived by multiplying two basic features: the *total minutes of daily service use* and the *total minutes of international service usage*, i.e., a multiplicative interaction. Table1 presents the comparison results obtained using PyCaret, demonstrating the performance improvement achieved by incorporating interaction features alongside the baseline features.

Language Model-based Feature Generation

- **SA Model Selection**: Various Sentiment Analysis (SA) kernels (KS_1–KS_n) were evaluated by comparing the effect of their corresponding generated features on classifier performance. These kernels included bert-base-uncased (Geetha and Renuka, 2021), roberta-base (Liu, 2019), distilbert-base-cased and distilbert-base-uncased (Sanh, 2019), nlptown-bert-base-multilingual-uncased-sentiment (Sahoo et al., 2023), and cardiffnlp-twitter-roberta-base-sentiment (Barbieri et al., 2020). As shown in Table2, the cardiffnlp-twitter-roberta-base-sentiment kernel achieved the best performance. Therefore, the feature generated using this kernel was selected as the candidate feature, derived from SA-based analysis, and was added to the initial dataset along with interaction features.
- **ETD Model**: Various Emotional Tone Detection (ETD) kernels (KE_1–KE_n) were also compared based on the performance impact of their generated features. These included nateraw-bert-base-uncased-emotion, cardiffnlp-twitter-roberta-base-emotion (Camacho-Collados et al., 2022), j-hartmann-emotion-english-roberta-large (Hartmann, 2022), and bhadresh-savani-bert-base-uncased-emotion. As shown in Table3, the bhadresh-savani-bert-base-uncased-emotion kernel achieved the best results. Hence, the feature derived from this kernel was chosen as the candidate feature extracted through ETD-based analysis to be incorporated into the dataset alongside the interaction features.
- **TM Model:** Prior to this stage, a domain expert identified several key churn-related topics and assigned corresponding weights to reflect their potential impact on customer churn decisions. These topics included 'billing issue' (0.7), 'technical support' (0.6), 'product inquiry' (0.3), 'service cancellation' (0.9), and 'complaint' (1.0). These topics capture potential dissatisfaction points during the customer journey, particularly in the post-purchase phase. Since customers may experience multiple issues, the approach considers not only the most relevant topic but also the second-most relevant topic for each instance.

  Topic Modeling (TM) kernels (KT_1–KT_n) were evaluated by comparing their generated feature impact on classifier performance. The tested models included distilbert-base-uncased (Sanh, 2019), facebook-bart-large-mnli, roberta-large-mnli (Liu, 2019), t5-large (Raffel et al., 2020), and xlnet-large-cased (Yang, 2019). According to Table 4, roberta-large-mnli outperformed other TM kernels in isolation. Furthermore, incorporating both the top and second-most relevant topics using roberta-large-mnli yielded better results than using the top topic alone, enhancing the model's ability to detect complex customer behavior patterns. Thus, features generated by this kernel were selected as the final TM-based features for integration with the enriched dataset.

Table 2: Comparison of classifiers performance using baseline features alone and in combination with language model-enhanced features across various Sentiment Analysis (SA) kernels (KS_1–KS_6), evaluated using PyCaret.

Model	AUC	Recall	Precision	F1
Baseline features				
GBC	0.8723	0.4970	0.7446	0.5894
XGBoost	0.8504	0.5005	0.6830	0.5752
LightGBM	0.8504	0.4823	0.6711	0.5567
Baseline features + bert-base-uncased (KS_1)				
GBC	0.8907	0.5731	0.7855	0.6593

XGBoost	0.8806	0.6533	0.7811	0.7087
LightGBM	0.8869	0.6313	0.7908	0.6995
Baseline features + roberta-base (KS_2)				
XGBoost	0.8843	0.6313	0.7954	0.6990
GBC	0.8955	0.5915	0.8225	0.6850
LightGBM	0.8873	0.6348	0.8058	0.7078
Baseline features + distilbert-base-cased (KS_3)				
GBC	0.8980	0.5672	0.8088	0.6600
XGBoost	0.8894	0.6254	0.7884	0.6935
LightGBM	0.8889	0.6067	0.7588	0.6708
Baseline features + nltpown-bert-base-multilingual-uncased-sentiment (KS_4)				
GBC	0.9071	0.5951	0.8279	0.6895
XGBoost	0.8767	0.6532	0.7927	0.7136
LightGBM	0.8854	0.6427	0.8193	0.7169
Baseline features + distilbert-base-uncased (KS_5)				
XGBoost	0.8916	0.6278	0.7877	0.6943
LightGBM	0.8966	0.6569	0.8174	0.7254
RF	0.9041	0.5869	0.8361	0.6842
Baseline features + cardiffnlp-twitter-roberta-base-sentiment (KS_6)				
GBC	0.9662	0.7306	0.8688	0.7884
ET	0.9574	0.7083	0.8749	0.7790
RF	0.9548	0.6906	0.8927	0.7743

Table 3: Comparison of classifiers' performance using baseline features alone and in combination with language model-enhanced features across various Emotional Tone Detection (ETD) kernels (KE1–KE4), evaluated using PyCaret.

Model	AUC	Recall	Precision	F1
Baseline features				
GBC	0.8723	0.4970	0.7446	0.5894
XGBoost	0.8504	0.5005	0.6830	0.5752
LightGBM	0.8504	0.4823	0.6711	0.5567
Baseline features + nateraw-bert-base-uncased-emotion (KE_1)				
XGBoost	0.9237	0.7085	0.7972	0.7446
LightGBM	0.9288	0.6899	0.7993	0.7348
GBC	0.9372	0.6824	0.8081	0.7347
Baseline features + cardiffnlp-twitter-roberta-base-emotion (KE_2)				
GBC	0.9364	0.6463	0.8831	0.7416
XGBoost	0.9171	0.6790	0.7915	0.7261
LightGBM	0.9165	0.6604	0.7974	0.7200
Baseline features + j-hartmann-emotion-english-roberta-large (KE_3)				
XGBoost	0.9165	0.6861	0.8320	0.7479
LightGBM	0.9230	0.6786	0.8325	0.7447
RF	0.9202	0.6601	0.8541	0.7403
Baseline features + bhadresh-savani-bert-base-uncased-emotion (KE_4)				
GBC	0.9355	0.7048	0.8231	0.7582
XGBoost	0.9148	0.7197	0.8012	0.7568
LightGBM	0.9183	0.7046	0.8092	0.7516

After observing improvement in the classifiers' performance due to various features added to the feature set, we decided to visualize the feature importance bar chart, as in Figure2, using SHapley Additive exPlanations (SHAP) (Lundberg and Lee, 2017), to get more insight into the most important features affecting the predictive performance of ML models. SHAP values are derived from game theory and provide a way to explain the output of ML models by attributing contributions to each feature.

Figure2 shows the top ten important features with the corresponding average impact on the model output magnitude. Among them, churn score (generated in the Intuitive Feature Generation stage), negative sentiment (from the Language Model-based Feature Generation stage), int plan usage (produced in the Generative AI-Enabled Interaction Feature Generation stage), Complaint as the first topic, anger emotion, technical support needs as the second topic, and service

cancellation as the second topic (all generated in the Language Model-based Feature Generation stage) are included, respectively. The Shap value-based plot is indicative of the fact that our feature engineering approach was effective in dealing with the customer churn problem.

Table4: Comparison of classifiers performance using baseline features alone and in combination with language model-enhanced features across various Topic Modeling (TM) kernels (KT1–KT6), evaluated using PyCaret.

Model	AUC	Recall	Precision	F1
Baseline features				
GBC	0.8723	0.4970	0.7446	0.5894
XGBoost	0.8504	0.5005	0.6830	0.5752
LightGBM	0.8504	0.4823	0.6711	0.5567
Baseline features + distilbert-base-uncased (KT_1)				
GBC	0.9074	0.5951	0.8246	0.6883
XGBoost	0.8767	0.6532	0.7927	0.7136
LightGBM	0.8854	0.6427	0.8193	0.7169
Baseline features + facebook-bart-large-mnli (KT_2)				
GBC	0.9582	0.7224	0.8261	0.7682
XGBoost	0.9518	0.7369	0.8116	0.7691
RF	0.9553	0.7114	0.8631	0.7735
Baseline features + roberta-large-mnli (KT_3)				
XGBoost	0.9489	0.7485	0.8284	0.7833
LightGBM	0.9509	0.7481	0.8113	0.7753
RF	0.9481	0.7185	0.8638	0.7823
Baseline features + t5-large (KT_4)				
GBC	0.9036	0.5706	0.7984	0.6611
XGBoost	0.8909	0.6070	0.7759	0.6660
LightGBM	0.8985	0.6069	0.7772	0.6762
Baseline features + xlnet-large-cased (KT_5)				
XGBoost	0.8910	0.5955	0.7752	0.6704
LightGBM	0.8946	0.5988	0.8078	0.6856
RF	0.8829	0.5623	0.8315	0.6649
Baseline features + Multi-TP-roberta-large-mnli (KT_6)				
LightGBM	0.9746	0.8251	0.8882	0.8537
ET	0.9723	0.7996	0.9083	0.8478
RF	0.9759	0.7849	0.9223	0.8458

Enriched Feature-based Classifier Selection

At this stage, all features generated in the previous sections, along with the initial features, are passed through a feature selection process before being fed into a PyCaret pipeline. Afterwards, the best model with the best predictive and classification performance is selected to be applied to further customer churn tasks. Table5 is illustrative of the classifiers' performance after adding the Normalized-weighted Churn Score to the feature set. As shown, Random Forest, Extra Trees Classifier, and Gradient Boosting Classifier were the best outperforming models.

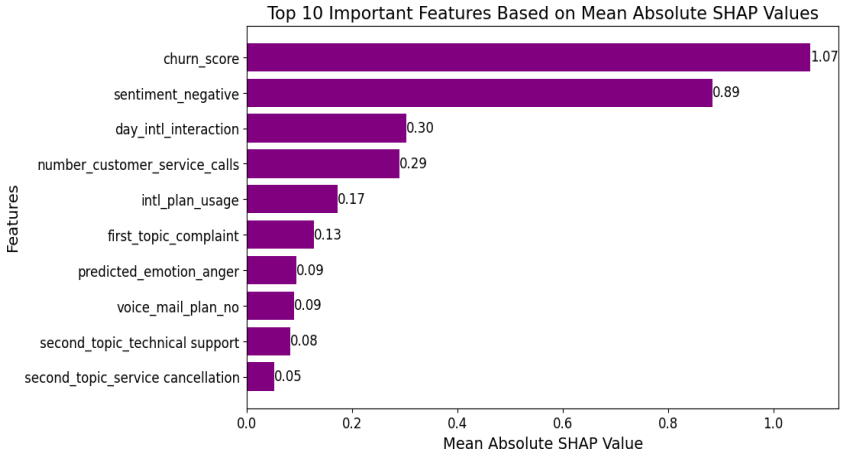

Figure 2: The feature importance bar chart, using SHAP to get more insight into the most important features affecting the predictive performance of ML models

Table 5: Comparison of classifiers performance using baseline features alone and in combination with features derived from leveraging all of the best kernels shown in Tables 2, Table 3, and Table 4+Intuitive feature generated in this study.

Model	AUC	Recall	Precision	F1
Baseline features				
GBC	0.8723	0.4970	0.7446	0.5894
XGBoost	0.8504	0.5005	0.6830	0.5752
LightGBM	0.8504	0.4823	0.6711	0.5567
Baseline features + Sentiment Analysis + Emotional Tone Detection + Topic Modelling + Intuitive feature generated in this study.				
GBC	0.9824	0.8319	0.8734	0.8501
RF	0.9828	0.8172	0.9210	0.8644
ET	0.9824	0.8060	0.9031	0.8503

Ablation Study

To observe the impact of factors contributing to model efficacy within our framework, we conducted an ablation study. Figure3 presents a comparative analysis of the F1-score, using the initial feature set as a baseline to assess the impact of both generated interaction features and LM-based features on classifier performance. The inclusion of all features, including initial features, interaction features, features from the best SA kernel, best ETD kernel, and best TM kernel (i.e., MTM), as well as the intuitively generated feature (Normalized-weighted Churn Score), resulted in superior classifier performance.

We systematically removed each generated feature and compared classifier performance. The removal of the intuitive feature, followed by LM-based features and then interaction features, resulted in a gradual decline in performance. In summary, this ablation study highlights the effectiveness of the feature engineering approach and demonstrates the value of interaction features, LM-based features, and the intuitively generated feature introduced in this study.

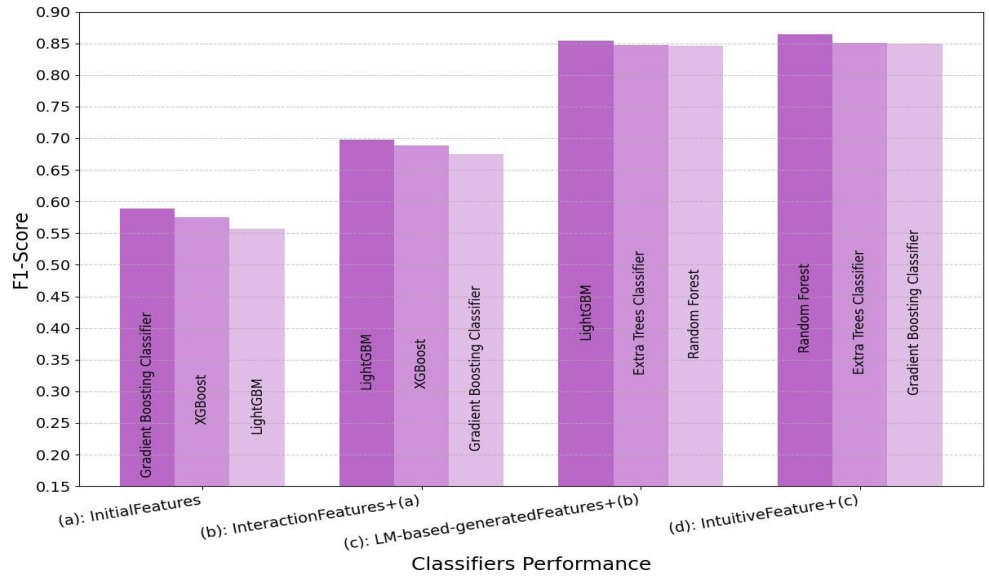

Figure 3: Comparative analysis of F1 scores in an ablation study using the initial feature set as the baseline to evaluate the impact of various generated LM-based features on churn classifier performance. The colors indicate descending F1-score values, respectively.

Conclusion

In this study, we proposed a language model-enhanced feature engineering framework for customer churn analysis, integrating textual data with domain expertise to improve predictive performance. By leveraging advanced techniques such as SA, ETD, and TM, we demonstrated how textual data can enrich the feature set and lead to better churn predictions. Our experiments showed a consistent improvement in classifiers performance as new features were added, starting with interaction features and progressing through sentiment- and topic-derived features. Notably, the introduction of a novel feature, i.e., Normalized-weighted Churn Score, further enhanced the model's predictive capabilities.

The results highlight the significant role of customer-generated textual data in conjunction with domain expertise in advancing feature engineering for customer churn analysis. This approach provides a comprehensive understanding of customer behavior by capturing not only structured data but also valuable unstructured ones. Our framework offers a promising solution for businesses aiming to improve churn prediction and, ultimately, customer retention strategies.

Future work could explore applying this framework across a broader range of industries and datasets and investigate integrating advanced NLP techniques to further refine feature extraction and improve predictive accuracy. Although this study is based on a single dataset, the framework's modular and domain-adaptable design, particularly the use of pre-trained language models, makes it well-suited for generalization to other churn scenarios with diverse textual characteristics, such as formal emails, product reviews, or social media posts, to assess its robustness and adaptability.

Acknowledgments

We acknowledge the Center for Applied Artificial Intelligence at Macquarie University (Sydney, NSW, Australia) and Prospa Advance Pty Limited (Sydney, NSW, Australia) for supporting and funding this research.

Endnotes

1. https://github.com/pycaret/pycaret
2. https://github.com/aws-samples/churn-prediction-with-text-and-interpretability/blob/main/README.md
3. This feature indicated the interaction between the number of customer service calls and day/evening minutes.
4. https://huggingface.co/nateraw/bert-base-uncased-emotion
5. https://huggingface.co/bhadresh-savani/bert-base-uncased-emotion
6. https://huggingface.co/facebook/bart-large-mnli

References

- Abou el Kassem, E., Hussein, S.A., Abdelrahman, A.M. and Alsheref, F.K. (2020) 'Customer churn prediction model and identifying features to increase customer retention based on user generated content,' International Journal of Advanced Computer Science and Applications, 11 (5).
- Agrawal, S., Das, A., Gaikwad, A. and Dhage, S. (2018), 'Customer churn prediction modelling based on behavioural patterns analysis using deep learning,' Proceedings of the 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE), IEEE, pp. 1–6.
- Alboukaey, N., Joukhadar, A. and Ghneim, N. (2020) 'Dynamic behavior based churn prediction in mobile telecom,' Expert Systems with Applications, 162, p. 113779.
- Barbieri, F., Camacho-Collados, J., Neves, L. and Espinosa-Anke, L. (2020), 'TweetEval: Unified benchmark and comparative evaluation for tweet classification,' arXiv preprint arXiv:2010.12421.
- Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003) 'Latent Dirichlet allocation,' Journal of Machine Learning Research, 3 (Jan), pp. 993–1022.
- Bobbitt, Z. (2022), 5 examples of nonlinear relationships between variables. [Online]. [Accessed May 9, 2025]. Available at: https://www.statology.org/nonlinear-relationship-examples/
- Breiman, L. (2001) 'Random forests,' Machine Learning, 45, pp. 5–32.
- Brooks, M., Amershi, S., Lee, B., Drucker, S.M., Kapoor, A. and Simard, P. (2015), 'FeatureInsight: Visual support for error-driven feature ideation in text classification,' Proceedings of the 2015 IEEE Conference on Visual Analytics Science and Technology (VAST), IEEE, pp. 105–112.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. et al. (2020) 'Language models are few-shot learners,' Advances in Neural Information Processing Systems, 33, pp. 1877–1901.
- Camacho-Collados, J., Rezaee, K., Riahi, T., Ushio, A., Loureiro, D., Antypas, D., Boisson, J., Espinosa-Anke, L., Liu, F., Martínez-Cámara, E. et al. (2022), 'TweetNLP: Cutting-Edge Natural Language Processing for Social Media,' Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Abu Dhabi, U.A.E.
- Chen, T. and Guestrin, C. (2016), 'XGBoost: A scalable tree boosting system,' Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794.
- De, S. and Prabu, P. (2022) 'Predicting customer churn: A systematic literature review,' Journal of Discrete Mathematical Sciences and Cryptography, 25 (7), pp. 1965–1985.
- Dias, J.R. and Antonio, N. (2023) 'Predicting customer churn using machine learning: A case study in the software industry,' Journal of Marketing Analytics, pp. 1–17.
- Friedman, J.H. (2001) 'Greedy function approximation: A gradient boosting machine,' Annals of Statistics, pp. 1189–1232.
- Geetha, M. and Renuka, D.K. (2021) 'Improving the performance of aspect-based sentiment analysis using fine-tuned BERT base uncased model,' International Journal of Intelligent Networks, 2, pp. 64–69.
- Geurts, P., Ernst, D. and Wehenkel, L. (2006) 'Extremely randomized trees,' Machine Learning, 63, pp. 3–42.
- Hartmann, J. (2022), Emotion English DistilRoBERTa-base. [Online]. [Accessed May 9, 2025]. Available at: https://huggingface.co/j-hartmann/emotion-english-distilroberta-base/
- Höppner, S., Stripling, E., Baesens, B., vanden Broucke, S. and Verdonck, T. (2020) 'Profit driven decision trees for churn prediction,' European Journal of Operational Research, 284 (3), pp. 920–933.
- Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T.Y. (2017) 'LightGBM: A highly efficient gradient boosting decision tree,' Advances in Neural Information Processing Systems, 30.

- Kenton, J.D.M.-W.C. and Toutanova, L.K. (2019), 'BERT: Pre-training of deep bidirectional transformers for language understanding,' Proceedings of NAACL-HLT, Vol. 1, Minneapolis, Minnesota.
- Knowles, C. (2021), Customer churn costing Australian businesses millions, report finds. [Online]. [Accessed May 9, 2025]. Available at: https://itbrief.com.au/story/customer-churn-costing-australian-businesses-millions-report-finds
- Lemon, K.N. and Verhoef, P.C. (2016) 'Understanding customer experience throughout the customer journey,' Journal of Marketing, 80 (6), pp. 69–96.
- Lghaouch, E.M., Ounacer, S., Ardchir, S. and Azzouazi, M. (2024) 'Enhancing sentiment analysis through topic modeling: Comprehensive overview,' In: Industry 5.0 and Emerging Technologies: Transformation Through Technology and Innovations, pp. 161–179.
- Liu, Y. (2019), 'RoBERTa: A robustly optimized BERT pretraining approach,' arXiv preprint arXiv:1907.11692.
- Lundberg, S. and Lee, S.-I. (2017), 'A unified approach to interpreting model predictions,' arXiv preprint arXiv:1705.07874.
- Ma, L. and Zhang, L. (2019), 'Multi-perspective feature generation based on attention mechanism,' Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), IEEE, pp. 1–8.
- Madanchian, M. (2024) 'Generative AI for consumer behavior prediction: Techniques and applications,' Sustainability, 16 (22), p. 9963.
- Mitrović, S., Baesens, B., Lemahieu, W. and De Weerdt, J. (2018) 'On the operational efficiency of different feature types for telco churn prediction,' European Journal of Operational Research, 267 (3), pp. 1141–1155.
- Mohammad, S.M. and Turney, P.D. (2013) 'Crowdsourcing a word–emotion association lexicon,' Computational Intelligence, 29 (3), pp. 436–465.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W. and Liu, P.J. (2020) 'Exploring the limits of transfer learning with a unified text-to-text transformer,' Journal of Machine Learning Research, 21 (140), pp. 1–67.
- Sahoo, A., Chanda, R., Das, N. and Sadhukhan, B. (2023), 'Comparative analysis of BERT models for sentiment analysis on Twitter data,' Proceedings of the 2023 9th International Conference on Smart Computing and Communications (ICSCC), IEEE, pp. 658–663.
- Sanh, V. (2019), 'DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter,' arXiv preprint arXiv:1910.01108.
- Seobility Wiki. (2021), Customer touchpoints. [Online]. [Accessed May 9, 2025]. Available at: https://www.seobility.net/en/wiki/Customer_Touchpoints
- Sifa, R., Bauckhage, C. and Drachen, A. (2014), 'The playtime principle: Large-scale cross-games interest modeling,' Proceedings of the 2014 IEEE Conference on Computational Intelligence and Games, IEEE, pp. 1–8.
- Slavchanyk, O., Fedushko, S., Mykhailyshyn, V., Shakhovska, N. and Syerov, Y. (2024), 'Artificial intelligence application for customer behavior and churn prediction,' In: Data-Centric Business and Applications: Advancements in Information and Knowledge Management, Volume 3, Springer, pp. 145–168.
- Slof, D., Frasincar, F. and Matsiiako, V. (2021) 'A competing risks model based on Latent Dirichlet Allocation for predicting churn reasons,' Decision Support Systems, 146, p. 113541.
- Steinberg, D. (2009), 'CART: classification and regression trees,' In: The Top Ten Algorithms in Data Mining, Chapman and Hall/CRC, pp. 193–216.
- Tamaddoni, A., Stakhovych, S. and Ewing, M. (2016) 'Comparing churn prediction techniques and assessing their performance: A contingent perspective,' Journal of Service Research, 19 (2), pp. 123–141.
- Tueanrat, Y., Papagiannidis, S. and Alamanos, E. (2021) 'Going on a journey: A review of the customer journey literature,' Journal of Business Research, 125, pp. 336–353.
- Verbeke, W., Dejaeger, K., Martens, D., Hur, J. and Baesens, B. (2012) 'New insights into churn prediction in the telecommunication sector: A profit driven data mining approach,' European Journal of Operational Research, 218 (1), pp. 211–229.
- Vo, N.N., Liu, S., Li, X. and Xu, G. (2021) 'Leveraging unstructured call log data for customer churn prediction,' Knowledge-Based Systems, 212, p. 106586.
- Wang, Y., Satake, K., Onishi, T. and Masuichi, H. (2018), 'Customer churn prediction using sentiment analysis and text classification of VOC,' In: Computational Linguistics and Intelligent Text Processing: 18th

International Conference, CICLing 2017, Revised Selected Papers, Part II 18, Springer, pp. 156–165.
- Wu, X., Li, P., Zhao, M., Liu, Y., Crespo, R.G. and Herrera-Viedma, E. (2022) 'Customer churn prediction for web browsers,' Expert Systems with Applications, 209, p. 118177.
- Yang, Z. (2019), 'XLNet: Generalized autoregressive pretraining for language understanding,' arXiv preprint arXiv:1906.08237.
- Zheng, A. and Casari, A. (2018) Feature engineering for machine learning: principles and techniques for data scientists, O'Reilly Media, Inc.