IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Call Center Call Propensity Optimization Using Artificial Intelligence: A Case Study

Kiran Sathyanarayana

AI consultant and career coach

Abstract—This case study dwells on the usage of Artificial intelligence techniques like speech to text and advanced machine learning to identify a problem related to high volume of calls coming to a call center related to a billing issue. The case study describes in detail the business problem , how the data sources were identified and selected, how the data preparation was done after data cleaning. Finally suitable machine learning algorithms were used to build the propensity models and get them ready for real time inference. The model was deployed on AWS Sagemaker as an end point. This model was used to generate the Next Best action(NBA) for the customer query

Keywords: Propensity Modelling,Call center, AI, Logistic Regression,Random Forest,Gradient Boosting,NBA(Next best action)

Index Terms - Component, formatting, style, styling, insert.

I. Introduction

Artificial intelligence plays an important role in optimizing call centre operations. The call centre of any big organization must optimize its workforce and systems to ensure smooth handling of calls and early resolution of customer enquiries to ensure customer satisfaction. Many companies are using traditional AI and generative techniques to optimize their call centre operations aimed at cost reduction and greater efficiency.

II. Background

A leading telecom company faced a significant challenge with the high number of enquiries to their call centre, particularly related to billing issues. The company wanted to cut down on the number of such calls by leveraging Artificial Intelligence technologies

III. Objective

The problem statement given was to develop a propensity model to predict the likelihood of a customer calling the call center due to billing issues based on their mobile app and website interactions. This model would enable the company to proactively address potential issues and provide Next Best Actions (NBA) to customers, thereby reducing call center volume.

IV. Literature Survey

Propensity models are statistical models used to estimate the probability that an individual will receive a particular treatment or intervention, given their observed characteristics. These models are particularly valuable in observational studies

Propensity models are generated based on different supervised machine learning methods. Based on the probability of an event of interest happening, a model can be used to classify a particular observation as a high, medium or a low propensity.
In this case study 3 machine learning models were used to detect the propensity, Logistic regression, random forests and extreme gradient boosting.

Logistic Regression is a machine learning algorithm which maximizes the log likelihood of the dependent variables or in other words, using a loss function which identifies the optimal set of weights. Once the probability scores are generated based on certain cutoff, the dataset can be classified as a low propensity, high propensity or medium propensity categories

Suppose we have a logistic regression model predicting whether a customer has a propensity to buy a product (1) or not (0) based on the amount of money spent on marketing ((X)):

logit(p)=−2+0.5X

Here, (beta_1 = 0.5). The odds ratio for (X) is:

$E^{(0.5)\approx}1.65$

This means that for every additional unit spent on marketing, the odds of a customer buying the product increase by approximately 65%

Another method used is Random Forests based propensity modelling. Random forests are an ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees. They are known for their high accuracy, ability to handle large datasets with higher dimensionality, and robustness to overfitting

Random Forest is an ensemble learning method used for classification, regression, and other tasks. It operates by constructing multiple decision trees during training and outputting the mode of the classes (classification) or mean prediction (regression) of the individual trees[1]. This comprises of the following steps

1. **Bootstrap Sampling**: Random Forest uses a technique called bootstrap sampling to create multiple subsets of the original dataset. Each subset is created by randomly selecting samples with replacement from the original dataset.
2. **Building Decision Trees**: For each subset, a decision tree is built. However, instead of considering all features for splitting nodes, Random Forest randomly selects a subset of features at each split. This process introduces diversity among the trees.
3. **Aggregation**: Once all the trees are built, the Random Forest makes predictions by aggregating the predictions of all

The other method used is Xtreme gradient boosting. **XGBoost** (Extreme Gradient Boosting) is a gradient boosting framework that uses a gradient descent algorithm to minimize a loss function. This loss function is defined as:

L(Φ) = ∑_{i=1}^n l(y_i, Φ(x_i)) + ∑_{k=1}^K Ω(f_k)

Where:

- **L(Φ)** is the overall loss function.
- **l(y_i, Φ(x_i))** is the loss for the i-th instance, where y_i is the true label and Φ(x_i) is the predicted value.
- **Ω(f_k)** is the regularization term for the k-th tree in the ensemble.
- **n** is the number of instances.
- **K** is the number of trees in the ensemble.
- **Regularization Term (Ω)** The regularization term helps prevent overfitting. It's typically a combination of L1 and L2 regularization:
  Ω(f_k) = λ∑_{j=1}^T (w_j)^2 + α∑_{j=1}^T |w_j| where:
  - **λ** is the L2 regularization parameter.
  - **α** is the L1 regularization parameter.
  - **T** is the number of leaves in the tree.

- **w_j** is the score associated with the j-th leaf.

XGBoost optimizes this objective function using a gradient descent algorithm. It calculates the gradient of the loss function with respect to the scores of each leaf and updates the scores accordingly.

In essence, XGBoost aims to find a set of trees that minimize the loss function while controlling overfitting through regularization.The following are the steps involved.

1. **Initialization:** A weak model (e.g., a shallow decision tree) is trained on the original dataset.
2. **Residual Calculation:** The errors or residuals between the predicted values and the actual target values are calculated.
3. **New Model Training:** A new weak model is trained to predict these residuals.
4. **Model Combination:** The predictions from the new model are added to the predictions from the previous models.
5. **Iteration:** Steps 2-4 are repeated multiple times, creating an ensemble of models.

Key Advantages of GBM:

- **High Performance:** GBM consistently achieves state-of-the-art results on various machine learning tasks.
- **Flexibility:** It can handle both regression and classification problems.
- **Robustness:** GBM is less prone to overfitting compared to other algorithms.
- **Interpretability:** The ensemble of decision trees can provide some insights into the model's decision-making process.

Hyperparameters in GBM:

- **Number of Estimators:** The number of weak models in the ensemble.
- **Learning Rate:** Controls the contribution of each new model to the overall prediction.
- **Max Depth:** The maximum depth of each decision tree.
- **Subsample:** The fraction of samples used for training each weak model.
- **Loss Function:** The function used to measure the error between predicted and actual values.

The impact of AI on call centre operations

Presenting a literature of usage of analytics and machine learning in call centre analytics. A report by Deloitte highlights that 81% of contact centre executives are investing in AI to enhance agent experience and operational efficiency[2]. The focus is on deploying self-service capabilities, modernizing infrastructure, and enabling agent technologies. Contact centers are using AI for agent-enabling technologies to improve the agent's experience and operational efficiency.

Conversational AI can be used to provide on-demand 24/7 service, simplify access to critical information, and personalize recommendations. 81% of respondents have invested in voice and text analytics(increased from 62% 2 years ago)

Contact centers are using voice/text analytics for Call/Contact Driver Analysis, insights for 'at risk' customers, and Service Quality improvement tracking and improvement

Research from Cornell University discusses how AI tools are transforming contact centre jobs, particularly in the US, Canada, Germany, and Norway[3]. The study examines the effects of AI on employment, skills, and working conditions, noting both positive and negative impacts.

A study published on SSRN explores the effects of voice-based AI systems in a large telecommunication company[4]. The findings indicate changes in call length, customer demand for human service, and customer complaints. An extensive review of NLP in contact centre automation uncovers research gaps and explores the benefits of transitioning to AI-driven natural language solutions[5]

V. SCOPE AND IMPACT OF THIS CASE

This case study outlines the impact of using cross channel datasets to understand our end consumer better. Though this case study has a telecom background, this case study can be used to model any cross-channel consumer behavior based on digital footprints of the customer

VI. DATASET

The following datasets were considered

- Customer Call Data: Historical records of customer calls to the call center, including call reasons, duration, and resolution status.
- Website Interaction Data: Logs of customer interactions on the company's website, such as page visits, time spent on billing pages, and actions taken (e.g., viewing bills, making payments).
- Mobile app usage data. Mobile app usage data such as time of opening and closing the app, the notification sent, the menu that the user clicked on etc.

VII. METHODOLOGY

1. **Identification of data sources:** The customer calls were transcribed using a speech to text conversion tool in real time and some critical call related entities were extracted specifically whether there was any calls related to billing issue within a duration of 20 minutes after the website or mobile app usage. The website metrics were captured and stored in a data warehouse. The first step involved identifying the right metrics and KPI's that need to be used. This involved discussion with subject matter experts and building a data dictionary. Initially there were about 170 features of which around 30 were shortlisted.

2. **EDA**: Exploratory Data Analysis(EDA) was conducted to understand the data distribution and identify any missing or inconsistent data. Data issues like missing values, duplicate values were treated to ensure data quality.

3. **Feature Engineering**: Relevant features were extracted from the website data, call centre data and mobile app data For example, the number of visits to the billing page, time spent on the billing section, and frequency of payment-related actions were considered as potential predictors.

4. **Model Development**: A machine learning model was developed to predict the propensity of a customer calling the call

   centre due to billing issues. Various algorithms were tested, including logistic regression, decision trees, and random forests. The model was trained and validated using historical data.

5. **Modelling Results**

   For Logistic Regression Coefficients (coef): These are the estimated coefficients for each feature.

Predictor Variable	Coefficient (β\betaβ)	Standard Error (SE)	Wald Statistic	p-value
Intercept	0.5	0.1	5.0	<0.001
Page Views	0.03	0.01	3.0	0.002
Bounce Rate	-0.02	0.005	-4.0	<0.001
Average Session Duration	0.04	0.02	2.0	0.045
Conversion Rate	0.1	0.03	3.33	0.001

Explaining some of the modelling results

- **Standard Error (**std err**)**: The standard error of the coefficients.
- **Wald Test Statistic (**z**)**: The ratio of the coefficient to its standard error.
- **p-value (**P>|z|**)**: The p-value for the Wald test. A small p-value (typically < 0.05) indicates that the coefficient is significantly different from zero.

The z-score tells us how many standard deviations the estimated coefficient is away from zero. A higher absolute value of the z-score indicates that the predictor is more significant.

The Wald test helps to assess whether the explanatory variables (predictors) in a logistic regression model significantly contribute to the model. Essentially, it tests the null hypothesis that a particular coefficient is equal to zero (i.e., the variable has no effect)

Metric	Value
Precision	0.75
Recall	0.65
F1 Score	0.69
Matthews Correlation Coefficient	0.41
AUC	0.96

Model accuracy metrics for Logistic regression

For Random forests algorithm, the various hyperparameters like number of features, the number of estimators, maximum depth were considered for building the model

Metric	Value
Precision	0.75
Recall	0.60
F1 Score	0.67
Matthews Correlation Coefficient	0.41
AUC	0.96

Model accuracy metrics for Random Forests

Metric	Value
Precision	0.8
Recall	0.60
F1 Score	0..7
Matthews Correlation Coefficient	0.6
AUC	0.90

Model Accuracy metrics for Gradient Boosting

The Winner model was gradient boosting which was deployed as a sagemaker end point

VIII. Model Deployment

1. **Web Session Data Capture**:
    - **User Interacts with Website**: Users interact with your website, generating web session data.
    - **Data Stored in Database**: This data is captured and stored in a database for further processing.
2. **API Gateway**:
    - **HTTP Request**: When a web session ends, an HTTP request is sent to the API Gateway.
    - **API Gateway**: Acts as a front door for all incoming requests, routing them to the appropriate Lambda function.
3. **Lambda Function**:
    - **Invoke Endpoint**: The Lambda function processes the incoming data and invokes the SageMaker endpoint.
    - **Real-Time Scoring**: The Lambda function sends the data to the SageMaker endpoint for real-time scoring.
4. **SageMaker Endpoint**:
    - **Model Inference**: The SageMaker endpoint performs model inference and returns the prediction.

IX. Real time Scoring Process

1. **Incoming Request**: When a web session ends, the session data is sent to the API Gateway.
2. **Data Preprocessing**: The Lambda function preprocesses the incoming data to match the format expected by the model.
3. **Model Inference**: The pre-processed data is sent to the SageMaker endpoint, which generates a prediction.
4. **Prediction Response**: The prediction is returned to the Lambda function, which then sends it back to the API Gateway.
5. **Next Best Action**: Based on the propensity scores, personalized NBAs were generated for customers. For example, customers with a high propensity score received proactive communication, such as emails or SMS, addressing potential billing issues and providing solutions

This architecture ensures that your model can score data in real-time, providing immediate insights and enabling timely actions based on the predictions.

X. OUTCOME

Reduction in Call Volume: The implementation of the propensity model and NBA strategy led to a significant reduction in the number of billing-related calls to the call centre by almost 5 percent

REFERENCES

[1] Breiman, L. (2001). Random forestsMachine Learning, 45(1), 5-32

[2] 2023 Global Contact Center Survey, https://www2.deloitte.com/content/dam/Deloitte/us/Documents/process-and-operations/us-consulting-trends-and-ai-in-contact-center.pdf

[3] AI in contact centers, https://ecommons.cornell.edu/server/api/core/bitstreams/a0ac9f50-5a22-4b3d-a9d9-2cc06824e31d/content

[4]Voice-based AI in Call Center Customer Service: A Natural Field Experiment, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3633100

[5]A review of natural language processing in contact centre automation,