Synthetic Customer Profiles for Product Recommendation

Client: Global Next Consulting India Pvt. Ltd.

Submitted By: Group 1

Aryan Poddar

Disha Deshmukh

Aditi Chaudhari

Himanshu Shakya

Lakshya Tomar

Mentor: Abhipsa Guha

Date of Submission: October 2025

Version: v1.0

Contents

List of Figures

List of Tables

Chapter 1

Executive Summary

E-commerce organizations rely heavily on customer data to personalize experiences, recommend relevant products, and forecast purchasing behavior. However, real-world datasets are often limited, incomplete, or constrained by privacy regulations, reducing the effectiveness of AI-based recommendation systems.

To overcome these challenges, developed AI-driven pipeline titled “Synthetic Customer Profiles for Product Recommendation.” This project focuses on generating privacy-preserving, statistically consistent, and behaviorally realistic customer data using CTGAN (Conditional Tabular GAN) and GPT-2 for textual enrichment.

The proposed pipeline comprises four technical stages:

1. Data Acquisition and Cleaning: Sourcing the dataset from Kaggle and preprocessing it through encoding, normalization, and missing-value handling.
2. Synthetic Data Generation: Employing CTGAN to produce realistic customer profiles that maintain authentic statistical relationships.
3. Exploratory Data Analysis (EDA): Comparing real and synthetic data through visual analyses such as pie, histogram, bar, and sunburst charts to assess data fidelity.
4. Machine Learning Modeling: Training and evaluating algorithms for product recommendation tasks.

Chapter 2

Introduction & Objectives

2.1 Problem Statement

In the e-commerce industry, personalized product recommendations play a crucial role in enhancing user experience and driving sales. However, developing effective recommendation systems often requires large volumes of high-quality customer data, which may not always be available due to privacy concerns, limited historical records, or new market scenarios.

The objective of this project is to generate synthetic customer profiles that mimic real-world purchasing behavior using CTGAN (Conditional Tabular GAN), thereby augmenting the available dataset. These synthetic profiles will be used to experiment with machine learning algorithms such as Random Forest, XGBoost, and LightGBM to predict and recommend products to users based on their demographic and behavioral attributes.

This approach aims to explore whether synthetic data can effectively support recommendation systems, ensuring robust, scalable, and privacy-preserving solutions for personalized product suggestions.

2.2 Project Objectives

The following objectives outline the key goals of this project, focusing on leveraging synthetic data to enhance the effectiveness, scalability, and reliability of e-commerce product recommendation systems:

1. Enable Privacy-Safe Data Generation: Create synthetic customer profiles that preserve user privacy while reflecting realistic purchasing behavior.

2. Enhance Recommendation Accuracy: Leverage the augmented dataset to improve the predictive performance of machine learning models, ensuring more relevant product suggestions.

3. Support Scalable Model Development: Develop a repeatable pipeline to generate synthetic data efficiently within 60 seconds, facilitating faster training and testing of recommendation models.

4. Maintain Data Fidelity: Ensure that the synthetic data accurately captures the statistical patterns and correlations present in the real dataset, supporting reliable model predictions.

Chapter 3

Dataset Details

The project utilized the Kaggle *Customer Shopping Trends Dataset*, comprising 3,900 records with 18 attributes representing both demographic and behavioral aspects of customers. De- spite its small size, the dataset provides a reliable foundation to evaluate CTGAN's capability to model complex dependencies.

3.1 Dataset Overview

Attribute	**Description**
Source	Kaggle
Format	CSV
Records	3,900
Features	18 (Demographic + Behavioral)

Table 3.1: Dataset Summary

3.2 Key Features

Demographic: Age, Gender, Location

Behavioral: Item Purchased, Category, Purchase Amount (USD), Frequency of Purchases, Subscription Status, Payment Method, Shipping Type, Discount Applied, Promo Code Used, Previous Purchases, Review Rating, Size, Color, Season

3.3 Preprocessing Steps

1. Removed duplicate records (if any) to ensure data consistency.

2. Detected and handled outliers in monetary fields using the Interquartile Range (IQR) method.

3. Filled missing values using median imputation for numerical features and mode imputation for categorical ones.

4. Encoded categorical features using Label Encoding for ordinal variables and One-Hot Encoding for nominal variables.

5. Standardized numerical attributes (Age, Review Rating, Previous Purchases, Purchase Amount) using StandardScalerto normalize data distribution.

Chapter 4

Project Architecture & Methodology

4.1 Technical Architecture

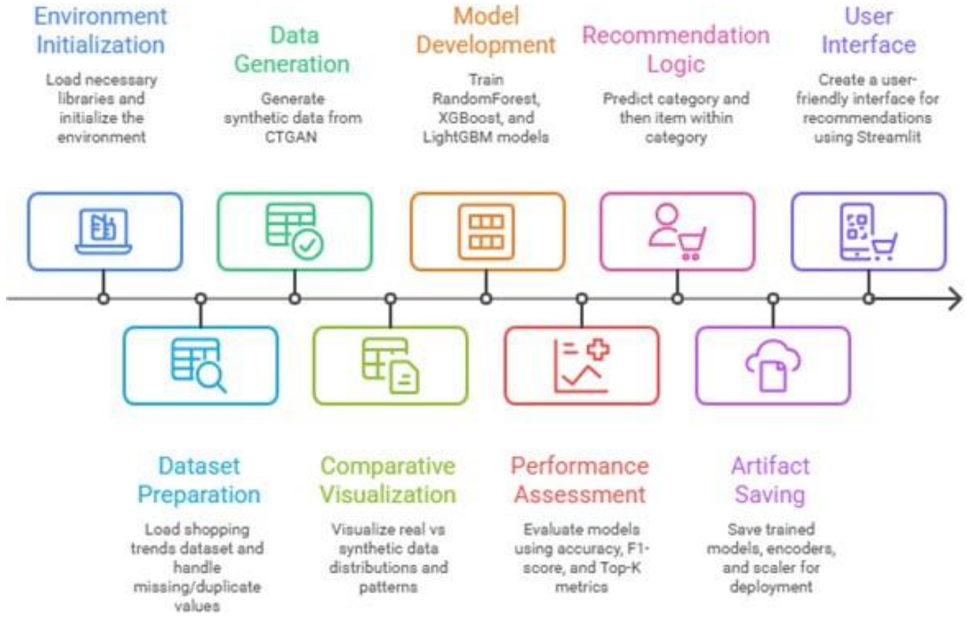
Environment Initialization
Load necessary libraries and initialize the environment
Data Generation
Generate synthetic data from CTGAN
Model Development
Train RandomForest, XGBoost, and LightGBM models
Recommendation Logic
Predict category and then item within category
User Interface
Create a user-friendly interface for recommendations using Streamlit
Dataset Preparation
Load shopping trends dataset and handle missing/duplicate values
Comparative Visualization
Visualize real vs synthetic data distributions and patterns
Performance Assessment
Evaluate models using accuracy, F1-score, and Top-K metrics
Artifact Saving
Save trained models, encoders, and scaler for deployment

Figure 4.1: End-to-End Pipeline of Recommendation System

4.1.1 Methodology

1. **Environment Initialization**

   Install and import required Python libraries such as pandas, numpy, sdv, scikit-learn, xgboost, lightgbm, and streamlit. Configure the working environment and set random seeds to ensure reproducibility.

2. **Dataset Preparation**

   Load the raw dataset from CSV or database, handle missing values, and remove duplicates. Encode categorical variables using Label Encoding or One-Hot Encoding. Normalize numerical features where necessary to maintain consistency.

3. **Synthetic Data Generation (CTGAN)**

   Train the CTGAN model on the cleaned dataset to generate synthetic customer profiles. Ensure sufficient synthetic records are produced to augment training data. Validate the generated data using statistical similarity metrics and visual inspection.

4. **Data Quality Assessment**

   Compare real and synthetic datasets through summary statistics and distribution visualizations. Verify that synthetic data preserves the statistical integrity and diversity of real-world data patterns.

5. **Model Development and Training**

   Split the combined dataset (real + synthetic) into training and testing sets. Develop multiple machine learning models—Random Forest, XGBoost, and LightGBM—and tune hyperparameters for optimal accuracy.

6. **Performance Evaluation**

   Evaluate the models using Accuracy, F1-Score, and Top-K Accuracy metrics. Select the best-performing model based on balanced performance across key indicators.

7. **Recommendation Generation**

   Use the selected model to predict customer preferences and generate product recommendations. Incorporate real-time user inputs (age, gender, season, location, etc.) for personalized results.

8. **Artifact Saving and Deployment**

   Serialize trained models, encoders, and scalers using pickle or joblib. Develop an interactive Streamlit interface for deployment, enabling real-time, data-driven recommendations for end users.

4.2 CTGAN Architecture

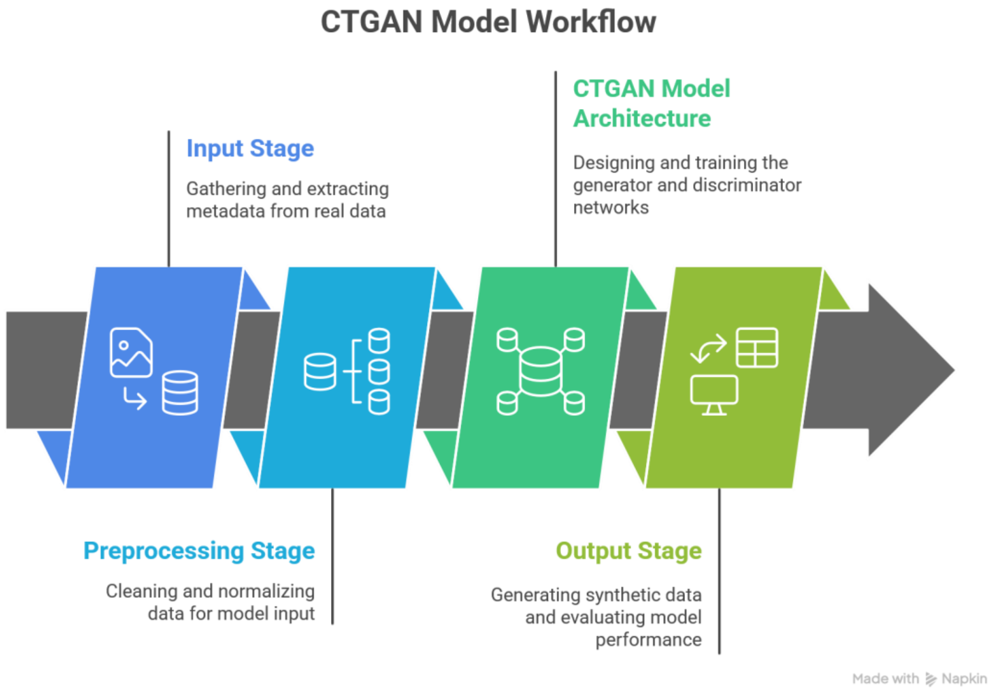
CTGAN Model Workflow
Input Stage
Gathering and extracting metadata from real data
CTGAN Model Architecture
Designing and training the generator and discriminator networks
Preprocessing Stage
Cleaning and normalizing data for model input
Output Stage
Generating synthetic data and evaluating model performance
Made with Napkin

Figure 4.2: CTGAN-based Synthetic Data Generation Architecture

CTGAN (Conditional Tabular GAN) generates realistic tabular data using a Generator and a Discriminator with conditional modeling for categorical features. The workflow can be summarized as follows:

1. **Prepare Data:** Preprocess real dataset and define categorical columns for conditional modeling.
2. **Noise Input:** Feed random noise and conditional vectors into the Generator.
3. **Generate Samples:** The Generator produces synthetic data samples.
4. **Discriminator Training:** Feed real and synthetic data to the Discriminator to clas- sify real vs fake.
5. **Generator Update:** Use Discriminator feedback to improve the Generator's output.
6. **Iterate Training:** Repeat Generator-Discriminator updates for multiple epochs until convergence.
7. **Output Synthetic Data:** Obtain high-fidelity synthetic tabular data that preserves statistical patterns of real data.

CTGAN (Conditional Tabular GAN) was used to generate synthetic customer profiles. Metadata and quality checks were included using SDV and SDD metrics.

Feature	**Description**
Model	CTGAN (Conditional Tabular GAN)
Library	SDV (Synthetic Data Vault)
Training Epochs	500
Input Data	Real customer records
Output	3,900 synthetic profiles with the same feature schema
Metadata	Used to capture feature distributions and dependencies
Quality Check	Evaluated using SDD metrics and quality report to ensure fidelity of synthetic data

Table 4.1: CTGAN Architecture, Metadata, and Quality Check Details

Chapter 5

Model Development

The models were trained on the Combined data (real + synthetic) to leverage the diversity and volume provided by the CTGAN. The primary task was a multi-class classification aimed at predicting the Target variable: Category and Item Purchased.

5.1 Random Forest Development

Workflow:

1. Preprocess numeric and categorical features using scaling and label encoding.
2. Split the combined dataset into training and testing sets.
3. Train Random Forest Classifier with 500 estimators, maximum depth 25, and balanced class weights.
4. Predict categories for the test set and evaluate using Accuracy and Top-3 Accuracy.

5.2 XGBoost Development

Workflow:

1. Preprocess features as done for Random Forest.
2. Train XGBoost Classifier with 500 estimators, maximum depth 10, and learning rate 0.1.
3. Predict categories for the test set and evaluate using Accuracy and Top-3 Accuracy.

5.3 LightGBM Development

Workflow:

1. Preprocess features as before and split dataset into train and test sets.
2. Train LightGBM Classifier for both category-level and item-level prediction:
   - Category Model: max depth 15, 500 estimators, learning rate 0.1
   - Item Models per category: max depth 10, 500 estimators, learning rate 0.1
3. Evaluate models using Accuracy and Top-3 Accuracy.
4. Apply gender-based filtering to ensure recommended items are appropriate for the predicted category.

Chapter 6

Model Evaluation

6.1 Model Summary

Metrics used to analyze the trained models are :

- **Accuracy:** Measures the overall correctness of predictions across all categories, calculated as the ratio of correctly predicted samples to the total number of samples.
- **F1-Score:** The harmonic mean of Precision and Recall, crucial for assessing performance on minority or less frequent categories, ensuring balanced evaluation.
- **Top-K Accuracy:** Evaluates whether the correct category is among the top K predicted categories, particularly useful in recommendation systems with multiple relevant suggestions.

Model	**Accuracy**	**Top-3 Accuracy**
Random Forest	0.4320	0.9059
XGBoost	0.4251	0.9038
LightGBM	0.4362	0.9059

Table 6.1: Comparison of Category Prediction Models

6.2 Visualizations

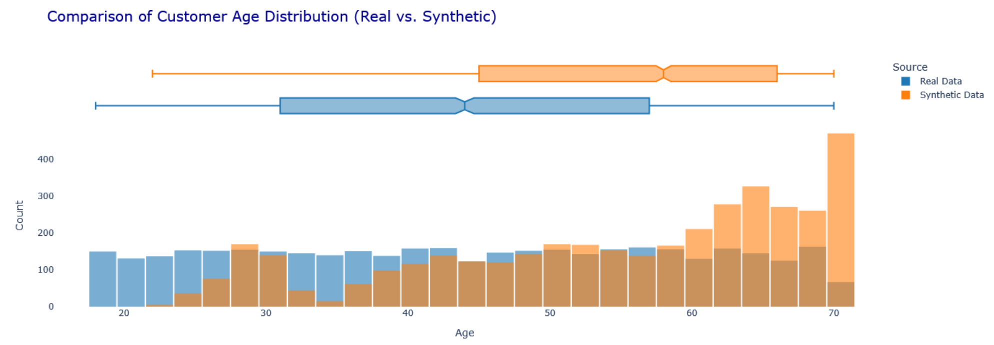

Figure 6.1: Age Distribution: Real vs Synthetic

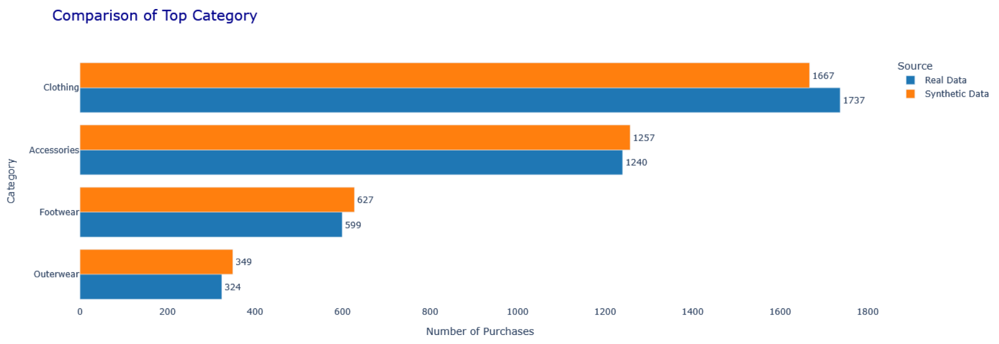

Figure 6.2: Top Customer Categories

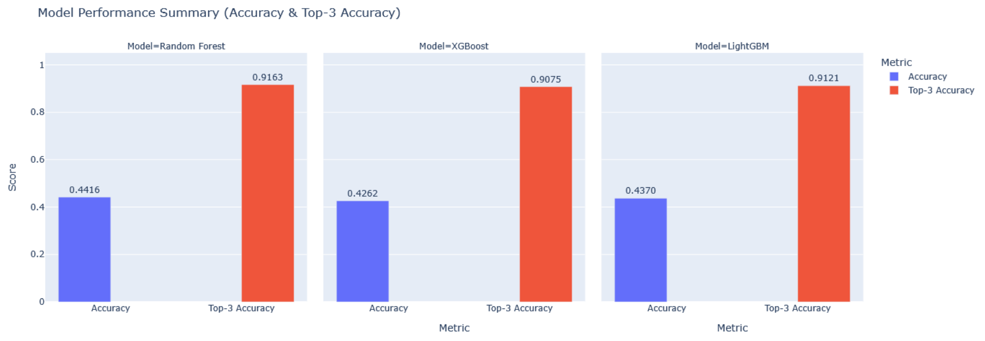

Figure 6.3: Model Performance Comparison

Chapter 7

Impact & Challenges

7.1 Business Impact

1. Enhances personalized product recommendations based on customer profiles.
2. Improves customer engagement and satisfaction by suggesting relevant items.
3. Increases sales and reduces churn through targeted recommendations.
4. Uses synthetic data to supplement limited real data for robust model training.
5. Provides actionable insights for business strategy and marketing decisions.

7.2 Challenges

1. High-cardinality target features caused very low exact accuracy for item-level predictions.
2. Data leakage during preprocessing or dataset combination inflated model performance metrics artificially.
3. Small real dataset size made the model prone to overfitting during hyperparameter tuning.
4. Balancing techniques and synthetic data augmentation sometimes led to overfitting.
5. Extensive feature engineering increased model complexity, making training and optimization more difficult.

Chapter 8

Conclusion

This project developed a recommendation system using both real and synthetic customer data, with CTGAN-generated profiles augmenting limited real data. Machine learning mod- els including Random Forest, XGBoost, and LightGBM were trained and evaluated, with Top-K Accuracy used as a practical metric for recommendations. The system demonstrates its ability to enhance personalized customer experiences, improve engagement, and sup- port business decisions. Challenges such as data imbalance, high-cardinality features, and overfitting were addressed through careful preprocessing and evaluation strategies. Future work can focus on deeper contextual modeling, real-time recommendations, and scalable deployment to maximize business impact.

Appendix A

Supplementary Information

A.1 Model Hyperparameters

Table A.1: Hyperparameter Summary

Model	Key Hyperparameters
Random Forest	n_estimators = 100, max_depth = 10, random_state = 42
XGBoost	learning_rate = 0.05, max_depth = 6, n_estimators = 200
LightGBM	num_leaves = 31, max_depth = -1, learning_rate = 0.05

A.2 Abbreviations

- **CTGAN** – Conditional Tabular Generative Adversarial Network
- **SDV** – Synthetic Data Vault
- **ML** – Machine Learning
- **UI** – User Interface
- **API** – Application Programming Interface