PREDICTING CUSTOMER SUPPORT TICKET TYPE USING MACHINE LEARNING AND NLP

Insights from Data analysis

Group 10

Chathurangi Akmeemana -16012

Kusara Udayana – 16073

Amindu Yohan - 16385

Abstract

This study aims to automate customer support ticket classification using **Machine Learning (ML)** and **Natural Language Processing (NLP)** to improve response efficiency and accuracy. Both structured data, such as customer demographics and ticket channels, and unstructured text data from ticket subjects and descriptions were analyzed. After preprocessing and feature engineering, **TF-IDF vectorization** was applied to extract meaningful text patterns, and several models including **Logistic Regression, Random Forest, Gradient Boosting, XGBoost**, and **LSTM networks** were trained. Traditional ML models showed limited predictive ability, while the **LSTM model** achieved better generalization without overfitting. Further experimentation with **BERTopic** provided insights into the semantic structure of customer issues, though it offered minimal improvement in accuracy. Overall, the project highlights the challenges of ticket classification with limited contextual features while demonstrating the potential of deep learning and NLP techniques in automating customer support systems.

Table of Contents

List of Figures

List of Tables

Introduction

In today's digital business environment, customer support plays a vital role in maintaining customer satisfaction and loyalty. Companies receive thousands of support tickets every day through various channels such as email, chat, and social media. These tickets may relate to technical problems, billing inquiries, or product-related questions. Traditionally, customer service agents manually review and classify these tickets to assign them to the correct department. However, this manual process is often slow, inconsistent, and prone to human error, which can lead to delays in issue resolution and reduced customer satisfaction.

With the growing volume of customer support requests, automation using **Machine Learning (ML)** and **Natural Language Processing (NLP)** has become an effective solution. By analyzing both structured data (such as product type and communication channel) and unstructured text data (such as ticket subject and description), machine learning models can automatically predict the type of ticket. This not only saves time but also ensures faster and more accurate tickets to the right support teams.

The purpose of this project is to develop a machine learning model that can automatically classify customer support tickets into predefined categories such as *technical issues*, *billing inquiry*, or *product inquiry*. By leveraging NLP techniques to process textual data and integrating additional structured information, the proposed system aims to improve ticket management efficiency, reduce response times, and enhance overall customer experience.

Description of the Question

1. **Identifying Key Factors Influencing Ticket Type**

   Determine which features (product, channel, priority, and text patterns) best differentiate ticket categories like Technical Issue, Billing Inquiry, and Product Inquiry.

2. **Building a Ticket Type Classification Model**

Develop a machine learning model that predicts Ticket Type using structured data (product, priority, channel) and text data (subject, description) to automate ticket routing and improve response efficiency.

Description of the Dataset

This project was conducted using a publicly available dataset found on Kaggle, titled "Customer Support Ticket Dataset (kaggle.com)". The dataset contains **8,469 records** and includes **17 variables**, out of which several represent structured customer and ticket details, and two variables **Ticket Subject** and **Ticket Description** provide unstructured text data. The target variable for this project is **Ticket Type**, which indicates the category of the customer's inquiry.

The description of each variable used in this project is presented in the table below.

Variable	Description
Ticket ID	A unique identifier for each ticket
Customer Name	The name of the customer who raised the ticket
Customer Email	The email address of the customer (domain name @example.com is used for privacy)
Customer Age	The age of the customer
Customer Gender	The gender of the customer
Product Purchased	The tech product purchased by the customer
Date of Purchase	The date when the product was purchased
Ticket Type	The type/category of the ticket (Technical Issue, Billing Inquiry, Product Inquiry, Refund request, Cancellation request)
Ticket Subject	The subject or topic of the ticket
Ticket Description	Description of the customer's issue or inquiry
Ticket Status	The current status of the ticket (e.g., Open, Closed, Pending)
Resolution	The resolution or solution provided for closed tickets
Ticket Priority	The priority level assigned to the ticket (e.g., Low, Medium, High, Critical)

Ticket Channel	The channel through which the ticket was raised (e.g., Email, Phone, Chat, Social Media)
First Response Time	The time taken to provide the first response to the customer
Time to Resolution	The time taken to resolve the ticket
Customer Satisfaction Rating	The customer satisfaction rating for closed tickets (scale of 1 to 5)

Table 1- Description of the Dataset

Data Pre-processing

Handling Missing Data and Feature Selection

The dataset was checked for duplicates and missing values. No duplicate records were found. During preprocessing, missing values were analyzed to ensure data quality. Variables such as **Resolution**, **Time to Resolution**, **First Response Time**, and **Customer Satisfaction Rating** had a high percentage of missing data (over 33–67%) because they were only available for resolved tickets. These columns, along with other non-predictive variables (Ticket ID, Customer Name, Customer Email, Date of Purchase) or post-resolution variables (Ticket Status, Ticket Priority), were removed to prevent **data leakage** and improve model reliability.

The final cleaned dataset retained only relevant features such as **Customer Age**, **Customer Gender**, **Product Purchased**, **Ticket Channel**, **Ticket Subject**, **Ticket Description** and **Ticket Type (Target)** for model development.

Handling Placeholders in Text

In the dataset, ticket descriptions contained placeholders such as **{product_purchased}**, representing the product name mentioned in the ticket. To make the text data more meaningful for analysis, a preprocessing function was created to automatically replace these placeholders with the actual product name from the **Product Purchased** column.

After this replacement, the **Product Purchased** column was removed since its information was already reflected in the text.

Text Cleaning Strategy

To prepare the text data for modeling, we implemented a balanced text preprocessing pipeline:

Text Processing Steps:

1. **Placeholder Replacement:** Replaced {product_purchased} with actual product names
2. **Lowercasing:** Converted all text to lowercase for consistency
3. **Pattern Removal:** Removed repetitive formal phrases (e.g., "I'm having an issue with")
4. **Monetary Symbol Replacement:**
   - Replaces all occurrences of the dollar sign (`$`) with the token `"money"`.
   - This preserves semantic meaning related to financial contexts (e.g., “money50 refund” indicates a refund discussion).
   - Unlike previous numeric normalization (`NUM` replacement), this approach keeps numeric values intact to retain useful quantitative cues.

5. **Stopword Removal:** Removed common English stopwords using NLTK
6. **Lemmatization:** Applied WordNet lemmatization for word normalization
7. **Custom Stopword Filtering:** Removed domain-generic terms (e.g., "issue", "please", "assist")

Key Decision: We deliberately retained domain-specific keywords such as "refund", "billing", "cancel" to preserve class-discriminative features, as these words are critical for distinguishing between ticket types.

Feature Engineering

We created two major feature sets: text-based (semantic) features and structured (categorical/numerical) features.

Text Features (TF-IDF Vectorization)

- **Method:** TF-IDF (Term Frequency–Inverse Document Frequency)
- **Parameters:**
  - Max Features: 300
  - N-gram Range: (1, 2)
  - Min Document Frequency: 3

  - Max Document Frequency: 0.8
  - Sublinear TF: True
- **Purpose:** Capture the importance of both single keywords and short phrases.

Structured Features

- **Categorical Variables:** Customer Gender, Ticket Subject, Ticket Channel → One-Hot Encoded.
- **Numerical Variable:** Customer Age → Standardized using StandardScaler.
- **Total Structured Features:** 24 dimensions.

Combined Feature Set

- Text Features: 300 dimensions
- Structured Features: 24 dimensions
- **Total Combined Features:** 324 dimensions

Important results of the descriptive analysis

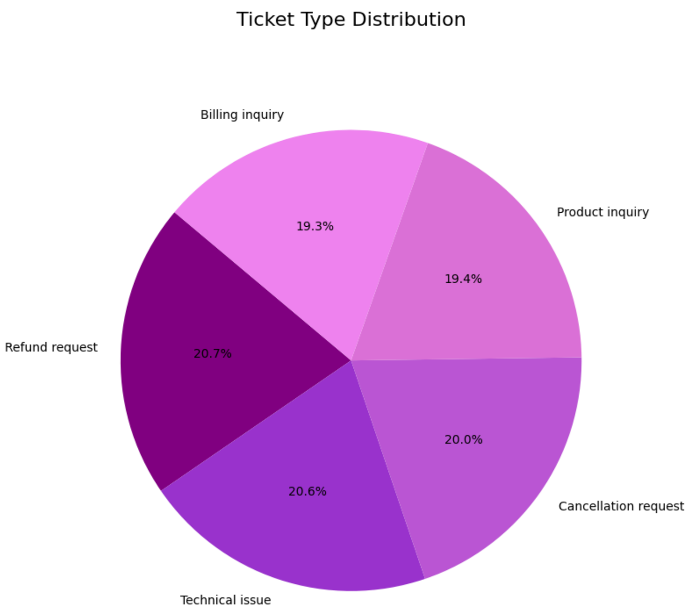

Figure 1 - Ticket Type Distribution

The dataset shows a relatively **balanced distribution** of ticket types. As seen in the pie chart, **Refund Requests (20.7%)** and **Technical Issues (20.6%)** are the most common categories, followed closely by **Cancellation Requests (20.0%)**, **Product Inquiries (19.4%)**, and **Billing Inquiries (19.3%)**.

This near-uniform distribution indicates that the dataset is **well-balanced**, with no major class imbalance problem. Such balance is beneficial for model training, as it helps prevent bias toward any single ticket type and ensures fair performance across all categories.

Important results of the advanced analysis

To better understand the performance and behavior of different models in ticket classification, multiple approaches were evaluated — including **traditional machine learning algorithms** and a **deep learning model**. The analysis aimed to identify which modeling strategy best captures the relationship between textual and structured features in support tickets.

Basic Modeling Approach

We began the modeling process with **traditional machine learning models** — *Logistic Regression*, *Random Forest*, and *Gradient Boosting Classifier* — under two feature configurations:

1. **Structured features only** (categorical and numerical information)
2. **Combined features** (structured + text-based TF-IDF embeddings).

However, since ticket descriptions contain rich textual information, models trained solely on structured data performed poorly (average accuracy below 19%). To capture contextual and linguistic variations, we also implemented a **Long Short-Term Memory (LSTM)** neural network, which learns sequential dependencies in text data.

Model Performance Summary

The performance of all models was evaluated using **accuracy**, **precision**, **recall**, and **F1-score**.

Model	Feature Configuration	Training Accuracy	Test Accuracy
Gradient Boosting	Combined	75.50%	21.02%
Random Forest	Combined	87.34%	20.90%
LSTM Network	Text Only	20.60%	20.63%
Gradient Boosting	Structured Only	53.25%	18.89%
Random Forest	Structured Only	70.55%	18.77%
Logistic Regression	Combined	28.46%	18.77%
Logistic Regression	Structured Only	22.54%	18.00%

Table 2 - Model Performance for basic models

Advanced Approach: Topic Modeling with BERTopic

As the TF–IDF–based models did not provide satisfactory results, we explored topic modeling to better capture the semantic meaning of customer inquiries. Using the **BERTopic** library, we generated topic representations for each inquiry description. These topics helped identify underlying themes in the text, making it easier to relate inquiries to specific ticket types.

We combined the topic features with additional variables such as **gender, age,** and **ticket channel** to create a richer feature set. Machine learning models were then trained on this enhanced dataset to predict the **ticket type** more accurately and effectively.

Model	Training Accuracy	Testing Accuracy
Logistic Regression	25%	19%
Random Forest	64%	22%
XGBoost	73%	21%

Table 3 - Model Performance for Topic Modeling with BERTopic

Among the trained models, **Logistic Regression** demonstrated the most balanced performance between training and testing accuracy, despite its lower overall score. In contrast, **Random Forest** and **XGBoost** achieved much higher training accuracies but performed poorly on the test set, indicating **overfitting**.

Since the initial topic modeling approach did not yield satisfactory results, we further utilized the **topic probability distributions** generated by BERTopic as additional numerical features. Each inquiry was represented by its probability scores across topics, which were then combined with other variables such as **gender** and **age**. Using this enhanced dataset, we trained machine learning models including **Random Forest** and **XGBoost**. However, the results still did not show significant improvement, indicating that even with topic probabilities, the models were unable to effectively capture the patterns needed to predict ticket types accurately.

To further improve performance, we also experimented with **fine-tuning a small language model** using only the **inquiry description** and corresponding **ticket type**. This approach aimed to directly capture contextual relationships between the text and its category. However, despite the model's ability to understand language semantics, the results did not show significant improvement, suggesting that additional contextual or structured information may be necessary for accurate ticket type prediction.

Issues you encountered and proposed solutions

1. **Issue**: All models achieved test accuracy between 18-21%, barely exceeding the random baseline of 20% for a balanced 5-class problem.
2. Issue: A key challenge was that several variables, such as **Resolution**, **Time to Resolution**, and **Customer Satisfaction Rating**, are only available after a ticket is processed. To avoid **data leakage**, these features had to be removed, leaving only a **limited set of variables** for training the models, which constrained predictive performance.

   Solution: To address the limited number of usable variables, we incorporated **BERTopic probability distributions** as additional features. These topic-based features, combined with other available variables like **gender** and **age**, were used to train machine learning models, providing richer input for predicting ticket types.

Discussion and conclusions

Best Model: LSTM

The **LSTM model** was selected as the best-performing approach, as it showed better generalization and was not overfitted despite having lower accuracy. **Hyperparameter tuning** was conducted using

KerasTuner Random Search, exploring different parameter combinations to improve model performance. The best configuration included an **embedding dimension of 256**, **64 LSTM units**, **64 dense layer units**, and **dropout rates of 0.3** after both the LSTM and dense layers. The model performed optimally with the **Adam optimizer** and a **learning rate of 0.00016**, which provided stable training and balanced model complexity with generalization.

Classification report for testing data for the LSTM model is given below.

Class	Precision	Recall	F1-score
Billing inquiry	0.19	0.2	0.2
Cancellation request	0.21	0.17	0.19
Product inquiry	0.16	0.15	0.15
Refund request	0.2	0.2	0.2
Technical issue	0.2	0.25	0.22

Table 4 - Classification Report

Despite experimenting with multiple modeling techniques—ranging from traditional machine learning using TF–IDF features to advanced topic modeling with BERTopic and fine-tuned language models—the results consistently showed limited predictive performance. The models struggled to accurately classify ticket types, indicating that the available data did not provide sufficient discriminatory information. In particular, variables such as **customer age**, **gender**, and **ticket channel** contributed minimally to improving model accuracy, while textual features like **subject** and **description** lacked the depth and contextual detail needed to effectively capture the nature of the issue.

Appendix

- Dataset Link - https://www.kaggle.com/datasets/suraj520/customer-support-ticket-dataset/data
- Codes - https://drive.google.com/drive/u/0/folders/1yBV4Z1LBP_IIDfhIK7rXztVPD6NVw4YT

References

- https://github.com/leylaeminova/Customer-Support-Ticket-Classification/tree/main
- https://sshivam-singh96.medium.com/automating-ticket-classification-with-nlp-nmf-and-machine-learning-a-step-towards-smarter-54ce3f3b6dc9