## horizontal line



## **LLM Classification Finetuning**

# Predicting User Preferences in Large Language Model Chatbot Responses

**By Tarun Penna,** [**penna.tarun@gmail.com**](mailto:penna.tarun@gmail.com)

# 

## **1. Introduction**

Large Language Models (LLMs) are increasingly integrated into conversational AI systems, yet ensuring their responses align closely with human preferences remains a significant challenge. This project focuses on developing a machine learning model that predicts which chatbot response a user will prefer in a head-to-head comparison, leveraging conversation data from Chatbot Arena.

By accurately modeling user preferences, this work contributes toward enhancing LLMs’ ability to generate responses that better meet user expectations and improve overall interaction quality.

## **2. Dataset Description**

The dataset employed consists of three components:

* **Training Set (train\_formatted.csv)**: Contains combined prompt and response texts, annotated with labels indicating the preferred response (A or B), or ties.
* **Test Set (test.csv)**: Contains prompts and two anonymous chatbot responses without preference annotations.
* **Sample Submission (sample\_submission.csv)**: Template file specifying the required format for predictions.

The training set includes 57,477 instances. For modeling purposes, tie cases were excluded to focus on definitive user preferences.

## 

## 

## 

## **3. Data Preprocessing**

Key preprocessing steps included:

* Filtering the training data to retain only samples with clear preferences (A or B).
* Mapping categorical labels to numerical values (A → 0, B → 1) to facilitate modeling.
* Combining prompts and corresponding responses into a single textual input field.
* Handling missing or empty entries by replacing them with empty strings.
* Applying analogous preprocessing to test data to ensure consistency.

## **4. Feature Engineering**

Textual data was transformed using **Term Frequency–Inverse Document Frequency (TF-IDF)** vectorization:

* The vectorizer was constrained to the top 5,000 features to balance feature richness and computational efficiency.
* This conversion enabled representation of text data as numerical feature vectors suitable for machine learning.

## 

## 

## 

## 

## **5. Modeling Approach**

The preference prediction model was implemented using **XGBoost**, a gradient boosting framework renowned for its performance on structured data:

* The dataset was split into training (90%) and validation (10%) subsets.
* The model was trained on TF-IDF feature vectors with corresponding labels.
* Evaluation metrics included accuracy, precision, recall, and F1-score.

## **6. Evaluation**

Model evaluation on the validation set demonstrated satisfactory predictive performance, indicating the approach’s effectiveness in discerning user-preferred responses based on textual input features.

## **7. Testing and Submission**

* The test dataset was vectorized with the TF-IDF model fitted on training data.
* Predictions were generated using the trained XGBoost classifier.
* Numerical predictions were mapped back to label format (A or B).
* Final predictions were saved in accordance with the sample submission template.

## 

## 

## 

## 

## 

## **8. Conclusion and Future Work**

This project successfully constructed a pipeline for predicting human preferences between chatbot responses, utilizing TF-IDF features and an XGBoost classifier. Potential directions to enhance this work include:

* Incorporating advanced language models or fine-tuning transformer architectures for deeper semantic understanding.
* Mitigating dataset biases, such as position or verbosity bias, to improve fairness and robustness.
* Conducting hyperparameter optimization and exploring alternate text representations.

## **9. Tools and Libraries**

* Python 3
* Pandas
* scikit-learn
* XGBoost

**Download Data set from :**

<https://www.kaggle.com/competitions/llm-classification-finetuning/data>