Project Report – Task 1: Smart Ticket Classification System

Internship Assignment - Vijayi WFH Technologies Pvt. Ltd.

# Objective

The purpose of this task was to build an intelligent, end-to-end machine learning pipeline capable of analyzing customer support tickets. Specifically, the goal was to:  
  
- Predict the issue type and urgency level of a ticket  
- Extract useful contextual information such as product names, complaint-related keywords, and any dates mentioned  
- Deploy the functionality through a user-friendly Gradio interface  
  
This simulated a real-world use case, where support teams benefit from AI that reduces manual workload and accelerates resolution times through automation and contextual insights.

# What Was Built

This task involved the complete development of a real-world NLP solution. The workflow consisted of the following stages:

|  |  |
| --- | --- |
| Step | Description |
| Data Preprocessing | Text cleaning, normalization, tokenization, lemmatization, and handling missing data |
| Feature Engineering | TF-IDF vectorization, length of ticket text, and sentiment score |
| Model Development | Two classifiers trained using Logistic Regression — one for issue type and another for urgency level |
| Entity Extraction | A rule-based approach to extract product names, dates, and complaint keywords |
| Integration | A single function that combines prediction and entity extraction |
| User Interface | An interactive Gradio app that provides input/output accessibility for any user |

# Output Snapshot

Input Example:

"My new refrigerator stopped cooling completely since yesterday. It was delivered on 15/06/2025. I need a technician urgently!"

Output (Displayed via Gradio UI):

- Predicted Issue Type: Product Defect  
- Predicted Urgency Level: High  
- Extracted Entities:  
 - Product: None  
 - Date: 15/06/2025  
 - Complaint Keywords: None

## Data Preprocessing

The support ticket text was cleaned and normalized using standard NLP techniques. This included converting to lowercase, removing punctuation and numbers, eliminating stopwords, and applying lemmatization. Any missing or malformed entries were handled carefully to ensure robustness.

## Feature Engineering

TF-IDF vectors, ticket length, and sentiment polarity were used to provide the models with rich input features.

## Model Development

Two logistic regression models were trained: one for classifying issue types and another for urgency levels. They were evaluated using classification reports.

## Entity Extraction

Rule-based extraction of products, dates, and complaint-related terms using regex and keyword matching.

## Integration

A single callable Python function integrates preprocessing, prediction, and entity extraction.

## Gradio Interface

A web-based user interface that allows anyone to interact with the model by entering raw text and viewing predictions in real-time.

# Conclusion and Reflections

This project demonstrated how classical machine learning and rule-based NLP can be combined to build an intelligent ticket classification and analysis system. It showcases the practical application of AI in customer service automation by delivering predictions that are not only accurate but also interpretable.  
  
Strengths of the solution:  
- Transparent and explainable model  
- Lightweight and efficient (no need for large-scale LLMs)  
- Deployable and user-friendly via the Gradio interface  
  
Areas for enhancement:  
- Expand entity detection with NER or embedding-based matching  
- Address cases with new/unseen complaint vocabulary  
- Improve robustness of sentiment analysis for ambiguous phrasing

# Deliverables Summary

|  |  |
| --- | --- |
| Deliverable | Status |
| Cleaned and documented notebook | Complete |
| Modular Python functions for each step | Complete |
| Trained models for issue type and urgency | Complete |
| Entity extraction module | Complete |
| Integrated analysis function | Complete |
| Gradio interface | Complete |
| Output demonstration screenshot | Attached |
| Humanized technical report | This document |
| Demo video walkthrough | Pending (to be recorded) |