Model Planning

Rujul Dwivedi

February 12, 2025

1 Introduction

In this document, we outline the model planning process for our AI-based retail product classification system. The goal is to accurately categorize products based on metadata, descriptions, and other features using machine learning (ML) techniques.

2 Classification Problem Definition

Given a dataset containing product descriptions, images, and features, the classification problem can be defined as:

$$f(X) \to Y \tag{1}$$

where:

- X represents input features (text, images, categorical data, etc.).
- Y represents the product category.
- $f(X)$ is the classification function we aim to learn.

3 Feature Engineering

Feature engineering plays a crucial role in improving classification accuracy. The key features considered are:

- **Text Features**: Processed using NLP techniques such as TF-IDF, Word2Vec, and BERT embeddings.
- **Metadata**: Brand, price, and attributes encoded as categorical or numerical features.
- **Image Features**: Extracted using deep learning models like ResNet or VGG.

4 Machine Learning Models Considered

Several classification models were considered for this problem:

- **Random Forest**: Robust and interpretable, handles categorical and numerical features well.
- **XGBoost**: Efficient for structured data, helps in reducing overfitting.
- **Support Vector Machine (SVM)**: Effective in high-dimensional spaces, but computationally expensive.
- **Deep Learning (CNN/RNN)**: Suitable for image-based and text-based classification.

5 Mathematical Representation

5.1 Random Forest

Random Forest consists of multiple decision trees:

$$Y = \frac{1}{N} \sum_{i=1}^{N} T_i(X) \tag{2}$$

where $T_i(X)$ is the output of individual trees.

5.2 XGBoost

XGBoost minimizes the following objective function:

$$\mathcal{L} = \sum_{i} l(y_i, \hat{y}_i) + \sum_{k} \Omega(f_k) \tag{3}$$

where l is the loss function and $\Omega(f_k)$ is the regularization term.

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.pipeline import Pipeline

# Sample Data
X_train, y_train = ["Product␣description␣1", "Product␣description
   ␣2"], ["Category␣A", "Category␣B"]

# Model Pipeline
pipeline = Pipeline([
    ('tfidf', TfidfVectorizer()),
    ('classifier', RandomForestClassifier(n_estimators=100))
])

# Training Model
pipeline.fit(X_train, y_train)
\print("Model␣trained␣successfully!")
```

6 Evaluation Metrics

To assess model performance, the following metrics are used:

- **Accuracy**: Measures overall correctness.
- **Precision**: Indicates correctness in positive predictions.
- **Recall**: Measures coverage of actual positives.
- **F1-Score**: Harmonic mean of precision and recall.

7 Conclusion

A range of classification models were considered, with Random Forest and XGBoost emerging as strong candidates for structured data. Deep learning models can be incorporated for additional improvements, particularly in handling images and complex text features.