

Model Development Phase Template

Date	12-10-2024
Team ID	LTVIP2024TMID25000
Project Title	SMS Spam Detection
Maximum Marks	10 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be included in the final submission through screenshots. The **model validation and evaluation report** will summarize the performance of the spam detection model using metrics such as **accuracy, precision, recall, F1-score, and confusion matrices**. Multiple models, including Naive Bayes, Decision Tree, and Logistic Regression, will be evaluated to select the best-performing model.

Initial Model Training Code (5 marks):

BUILDING THE COUNT VECTORS WITH CountVectorizer

```
[ ] # Initialize CountVectorizer
    vectorizer = CountVectorizer(stop_words='english')

    # Fit and transform the cleaned text data
    X = vectorizer.fit_transform(df['cleaned_text'])

    # Convert the label into binary format
    y = df['label_num']
    #print(y) optional to check the required label is displaying or not
```

SPLITTING THE DATA INTO TRAINING AND TESTING SETS

```
[ ] X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

MODEL BUILDING (NAVIE BAYES)

```
# Initialize and train the model
model = MultinomialNB()
model.fit(X_train, y_train)

# Make predictions
y_pred = model.predict(X_test)
```

Model Validation and Evaluation Report (5 marks):

Model	F1 Score	Confusion Matrix
Naive Bayes	97.58%	<pre># Confusion Matrix cm = confusion_matrix(y_test, y_pred) print("Confusion Matrix:") print(cm)</pre> <p>Confusion Matrix: [[730 12] [13 280]]</p>
Decision Tree	94.63%	<pre># Confusion Matrix cm = confusion_matrix(y_test, y_pred) print("Confusion Matrix:") print(cm)</pre> <p>Confusion Matrix: [[730 12] [18 274]]</p>