Text Classification using Logistic Regression

Last Updated : 04 Apr, 2025

Text classification is a fundamental task in Natural Language Processing (NLP) that involves assigning predefined categories or labels to textual data. It has a wide range of applications, including spam detection, sentiment analysis, topic categorization, and language identification.

Logistic Regression Working for Text Classification

Logistic Regression is a statistical method used for binary classification problems and it can also be extended to handle multi-class classification. When applied to text classification, the goal is to predict the category or class of a given text document based on its features. Below are the steps for text classification in logistic regression.

1. Text Representation:

- Before applying logistic regression text data should be converted as numerical features known as text vectorization.
- Common techniques for text vectorization include Bag of Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF), or more advanced methods like word embeddings (Word2Vec, GloVe) or deep learning-based embeddings.

2. Feature Extraction:

- Once data is represented numerically, these representations can be used as features for model.
- Features could be the counts of words in BoW, the weighted values in TF-IDF, or the numerical vectors in embeddings.

3. Logistic Regression Model:

- Logistic Regression models the relationship between the features and the probability of belonging to a particular class using the logistic function.
- The logistic function (also called the sigmoid function) maps any real-valued number into the range [0, 1], which is suitable for representing probabilities.
- The logistic regression model calculates a weighted sum of the input features and applies the logistic function to obtain the probability of belonging to the positive class.

Logistic Regression Text Classification with Scikit-Learn

We'll use the popular SMS Collection Dataset, consists of a collection of SMS (Short Message Service) messages, which are labeled as either "ham" (non-spam) or "spam" based on their content. The implementation is designed to classify text messages into two categories: spam (unwanted messages) and ham (legitimate messages) using a logistic regression model. The process is broken down into several key steps:

Step 1. Import Libraries

The first step involves importing necessary libraries.

- Pandas is used for data manipulation.
- **CountVectorizer** for converting text data into a numeric format.
- Various functions from **sklearn.model_selection** and **sklearn.linear_model** for creating and training the model.
- functions from **sklearn.metrics** to evaluate the model's performance.

```
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, confusion_matrix
```

Step 2. Load and Prepare the Data

- Load the dataset from a CSV file and rename columns for clarity.

- **Map labels from text to numeric values (0 for ham, 1 for spam)**, making it suitable for model training.

```
data = pd.read_csv('/content/spam.csv', encoding='latin-1')
data.rename(columns={'v1': 'label', 'v2': 'text'}, inplace=True)
data['label'] = data['label'].map({'ham': 0, 'spam': 1})
```

Step 3. Text Vectorization

Convert text data into a numeric format using CountVectorizer, which transforms the text into a sparse matrix of token counts.

```
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(data['text'])
y = data['label']
```

Step 4. Split Data into Training and Testing Sets

Divide the dataset into training and testing sets to evaluate the model's performance on unseen data.

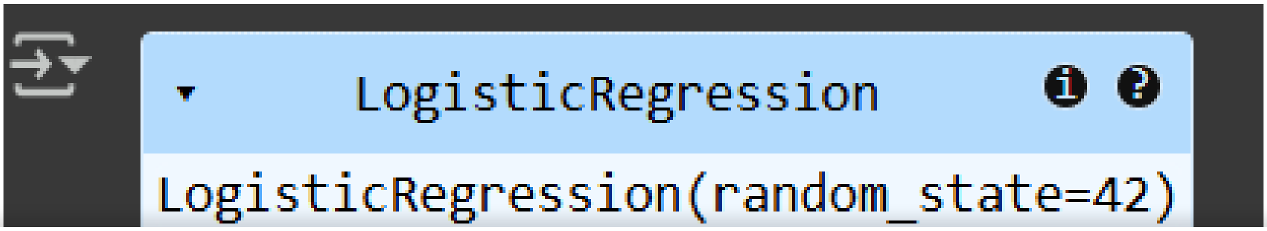
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25,
random_state=42)
```

Step 5. Train the Logistic Regression Model

Create and train the logistic regression model using the training set.

```
model = LogisticRegression(random_state=42)
model.fit(X_train, y_train)
```

Output:



```
LogisticRegression(random_state=42)
```

Logistic Regression

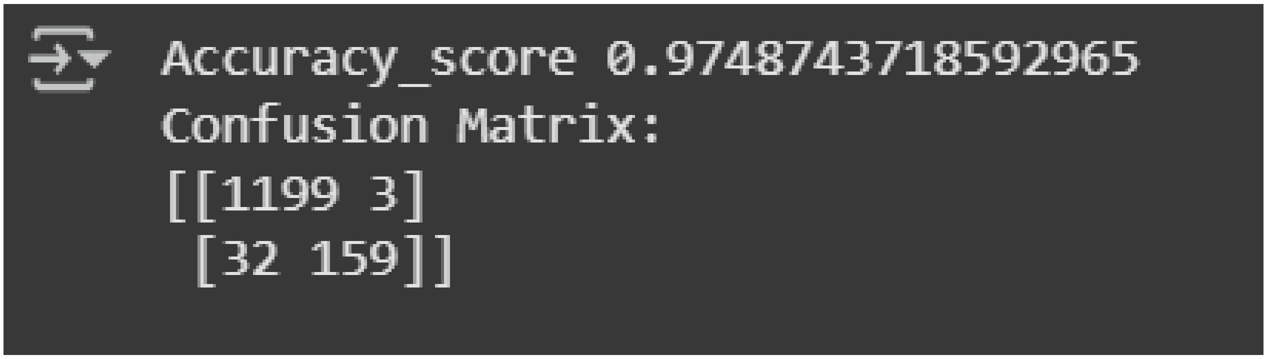
Step 6. Model Evaluation

Use the trained model to make predictions on the test set and evaluate the model's accuracy and confusion matrix to understand its performance better.

```
y_pred = model.predict(X_test)
print("Accuracy_score" ,accuracy_score(y_test, y_pred))

cm = confusion_matrix(y_test, y_pred)
print("Confusion Matrix:")
print(f"[[{cm[0,0]} {cm[0,1]}]")
print(f" [{cm[1,0]} {cm[1,1]}]]")
```

Output:



```
Accuracy_score 0.9748743718592965
Confusion Matrix:
[[1199 3]
 [32 159]]
```

The model is 97.4% correct on unseen data. The **Confusion Matrix** stated:

- 1199 messages correctly classified as 'ham'.
- 159 messages correctly classified as 'spam'.
- 32 'ham' messages wrongly labeled as 'spam'
- and 3 'spam' wrongly labeled as 'ham'.

Step 7. Manual Testing Function to Classify Text Messages

To simplify the use of this model for predicting the category of new messages we create a function that takes a text input and classifies it as spam or ham.

```
def classify_message(model, vectorizer, message):
    message_vect = vectorizer.transform([message])
    prediction = model.predict(message_vect)
    return "spam" if prediction[0] == 0 else "ham"
```

Output:

spam

This function first vectorizes the input text using the previously fitted CountVectorizer then predicts the category using the trained logistic regression model, and finally returns the prediction as a human-readable label.

This experiment demonstrates that logistic regression is a powerful tool for classifying text even with a simple approach. Using the SMS Spam Collection dataset we achieved an impressive accuracy of 97.6%. This shows that the model successfully learned to distinguish between spam and legitimate text messages based on word patterns.

Comment More info Advertise with us

Next Article

Text Classification using HuggingFace Model

Similar Reads

ML | Why Logistic Regression in Classification ?

Using Linear Regression, all predictions >= 0.5 can be considered as 1 and rest all < 0.5 can be considered as 0. But then the question arises why classification can't be performed using it? Problem - Suppose we are...

3 min read

ML | Logistic Regression v/s Decision Tree Classification

Logistic Regression and Decision Tree classification are two of the most popular and basic classification algorithms being used today. None of the algorithms is better than the other and one's superior performance ...

2 min read

Text classification using CNN

Text classification is a widely used NLP task in different business problems, and using Convolution Neural Networks (CNNs) has become the most popular choice. In this article, you will learn about the basics of...

5 min read

A Classification and Regression Tree(CART) is a Machine learning algorithm to predict the labels of some raw data using the already trained classification and regression trees. Initially one needs enough labelled data to...

4 min read

Text Classification using HuggingFace Model

Text classification is a pivotal task in natural language processing (NLP) that categorizes text into predefined categories. It is widely used in sentiment analysis, spam detection, topic labeling, and more. The developmen...

3 min read

Text Classification using Decision Trees in Python

Text classification is the process of classifying the text documents into predefined categories. In this article, we are going to explore how we can leverage decision trees to classify the textual data. Text Classification and...

5 min read

Logistic Regression using Statsmodels

Prerequisite: Understanding Logistic RegressionLogistic regression is the type of regression analysis used to find the probability of a certain event occurring. It is the best suited type of regression for cases where we...

4 min read

Weighted logistic regression in R

Weighted logistic regression is an extension of logistic regression that allows for different observations to contribute differently to the estimation process. This is particularly useful in survey data where each...

4 min read

Logistic Regression Vs Random Forest Classifier

A statistical technique called logistic regression is used to solve problems involving binary classification, in which the objective is to predict a binary result (such as yes/no, true/false, or 0/1) based on one or more...

7 min read

Logistic regression vs clustering analysis

Data analysis plays a crucial role in various fields, helping organizations make informed decisions, identify trends, and solve complex problems. Two widely used methods in data analysis are logistic regression and...

7 min read

A-143, 7th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305)

Registered Address:
K 061, Tower K, Gulshan Vivante
Apartment, Sector 137, Noida, Gautam
Buddh Nagar, Uttar Pradesh, 201305

Company

About Us
Legal
Privacy Policy
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
Top 100 DSA Interview Problems
DSA Roadmap by Sandeep Jain
All Cheat Sheets

Web Technologies

HTML
CSS
JavaScript
TypeScript
ReactJS
NextJS

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial
Tutorials Archive

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning
ML Maths
Data Visualisation
Pandas
NumPy
NLP
Deep Learning

Python Tutorial

Python Programming Examples
Python Projects
Python Tkinter
Python Web Scraping
OpenCV Tutorial
Python Interview Question

Computer Science

Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths
Software Development
Software Testing

DevOps

Git
Linux
AWS
Docker
Kubernetes
Azure
GCP
DevOps Roadmap

System Design

High Level Design
Low Level Design
UML Diagrams
Interview Guide
Design Patterns
OOAD
System Design Bootcamp
Interview Questions

Inteview Preparation

Competitive Programming
Top DS or Algo for CP
Company-Wise Recruitment Process
Company-Wise Preparation
Aptitude Preparation
Puzzles

School Subjects

Mathematics
Physics
Chemistry
Biology
Social Science
English Grammar
Commerce
World GK

GeeksforGeeks Videos

DSA
Python
Java
C++
Web Development
Data Science
CS Subjects