Address Classification

Problem Statement:-

The Main Aim of this assignment is to create a model that can be able to classify the given address as commercial / residential

Dataset Used:-

Dataset given by the delhivery

It has Below Attributes:

Id - Represents the index of the instance

Masked_address: It is address type by a particular user

Label :-

it is label for address, it has two unique values one is Commercial(addresses like company,office,hospital etc), residential(home addresses)

Problems in the Dataset:

1) Dataset contains almost more than 10k records in which target column named label has only 21 non-missing values labelled as commercial and residential, remaining all values are missing values
2) Now we can label the missing values in the dataset in order to develop a model

Approach Followed To solve this Problem Statement:

1) Analyzed the both Available labelled Commercial and residential data and extracted some patterns from them.

2) Addresses starts with door no,house,flat no,street,home are the residential addresses

3) Addresses starts with starts or contains strings like pvt,ltd,company,office,hospital etc are commercial addresses

4) Based on the above analysis i have labelled the all missing values as commercial or residential

5) Processed the text using regular expressions

6) Represented the text using Tf-IDF technique

7) Trained the dataset using ANN(Artificial Neural Networks)

8) Evaluated the Performance of the model using evaluation metrics like F1_score,precsion_score,recall,accuracy_score

Why??:-

1) Why i used simple ANN instead using popular algorithms like Rnn,Lstm,Gru which processes textual data very well?

Ans:

Rnn,lstm,Gru is used when there is Sequential information is to be preserved but here in our case in addresses there is not a sequential information in addresses every word is independent of other . that's why i didn't leveraged complex architectures like RNN,LSTM or any algorithms to process the data.

2)Why not Representing the text using Word2vec like Techniques:

Ans:

Word2vec is used when you want to represent your words in high dimensional vectors nothing but Word Embeddings this is specially used when there is a need of perseverance of sequential information. Here addresses is not a sequential information. That’s why i used Tf-idf instead of Word2vec.

Step by Step by Walkthrough Through the Pipelines Followed in Developing this model

1. Introduction

1.1 Purpose

The purpose of this comprehensive documentation is to provide detailed insights into the development, training, and evaluation of the Address Classification System. This system is designed to categorise addresses as either residential or commercial based on their attributes.

1.2 Scope

This document outlines the entire lifecycle of the Address Classification System, starting from data acquisition and preprocessing to the deployment of the trained model. It serves as a guide for developers, stakeholders, and users interested in understanding the system's functionality and processes.

1.3 Overview

The Address Classification System utilizes Deep learning and Nlp learning techniques to automate the categorization of addresses. It involves steps such as data setup, preprocessing, model development, evaluation, and deployment. Each section of this document delves into the specifics of these processes.

2. Data Setup

2.1 Importing Libraries

The initial step involves importing the necessary libraries required for data manipulation, text processing, and model development. This section ensures that the tools needed for subsequent stages are available.

2.2 Loading Data

The data setup phase includes loading the dataset containing masked addresses and labels. Pandas is used to read the data from a CSV file, laying the foundation for subsequent preprocessing.

3. Data Preprocessing

3.1 Converting to Lowercase

In this step, the addresses are converted to lowercase to ensure consistency in text processing. This eliminates discrepancies due to letter case variations.

3.2 Handling Null Values

Addresses with missing or null values are addressed using appropriate techniques. This stage ensures the dataset is ready for further processing without missing values.

3.3 Labelling Addresses

Addresses are classified as residential or commercial through labelling. This involves pattern matching and keyword analysis to assign appropriate labels to each address.

3.4 Text Processing

Text processing includes multiple operations such as replacing special characters and removing non-alphanumeric characters. This enhances the quality of text data for subsequent analysis.

3.5 Label Conversion

Labels are converted from string format to numerical values for compatibility with machine learning algorithms. This conversion enables seamless model training.

4. Model Development

4.1 ANN Model Architecture

The model development phase involves designing an Artificial Neural Network (ANN) model using the Keras library. The architecture consists of input, hidden, and output layers, forming the basis for classification.

4.2 Model Compilation and Training

The designed ANN model is compiled with appropriate optimizers and loss functions. It is then trained using the preprocessed data. This stage emphasises model convergence and accuracy.

5. Model Evaluation

5.1 Confusion Matrix

Model evaluation entails using a confusion matrix to visually represent model predictions against actual labels. This matrix provides insights into classification performance.

Confusion Matrix for actual and predicted values by the model is

array([[1532, 0],
[9, 1492]])

5.2 Metrics Overview

Key performance metrics such as F1 score, accuracy, precision, and recall are computed. These metrics quantify the model's effectiveness in address classification.

These are the Achieved Scores for Test Data:

f1_score of the model is 0.9969929836284664
accuracy_score of the model is 0.9970326409495549
precision_score of the model is 1.0
recall_score of the model is 0.9940039973351099

6. Model Saving

6.1 Component Saving

The trained model's components, including the vectorizer and model architecture and model weights , are saved for future use. This step ensures that the trained model can be reused without retraining.

7. Conclusion

7.1 Summary

In summary, the Address Classification System demonstrates the successful application of machine learning techniques to automate address categorization. It showcases the benefits of accurate classification for various applications and it should be very useful in Retail Application