Machine Learning Engineer Nanodegree
Capstone Proposal

Doyin Olarewaju
February 4th, 2020

Domain Background

Language is a method of human communication, either spoken or written, consisting of the use of words in a structured and conventional way. Human language is how we humans communicate, relay information, express desires, dissatisfaction and so much more. It is pretty easy for humans to understand, but for computers, not so much.

Natural Language Processing (shortened as NLP), is an area of artificial intelligence which deals with the interaction between computers and the human language. The ultimate objective of NLP is to read, decipher, understand, and make sense of the human languages in a manner that is valuable. Most NLP techniques rely on machine learning to derive meaning from human languages. There are several reasons why one might want to teach a computer to understand human language, some of which are:

- Language translation applications such as Google Translate
- Word Processors such as Microsoft Word and Grammarly that employ NLP to check the grammatical accuracy of texts.
- Personal assistant applications such as OK Google, Siri, Cortana, and Alexa.

What makes NLP so difficult? It is the nature of the human language. The rules that dictate the passing of information using natural language are not easy for computers to understand. For example, we could have sarcastic expressions whose meaning is not literal but can only be understood from the context. Understanding the human language requires understanding the words and how the concepts are connected to deliver the message.

Problem Statement

The main objective of this project is to recognize tweets from different people. We train the model to recognize tweet patterns of different people, then we give the model a tweet and it helps classify if the model is from one person or another to a degree of probability.

Datasets and Inputs

For this project, I will be mining my own dataset from twitter. The dataset will contain hundreds of tweets from two twitter accounts:

- Prosper Otemuyiwa (@unicodeveloper)
- John McAfee (@officialmcafee)

The dataset contains the following information:

- Serial Number
- Date tweet was created (date_created)
- Twitter handle of the originating account (handle)
- The tweet (text)
- Tweet Id (tweet_id)

The data will be prefetched and stored in a file, then read and used to train the model.

Solution Statement

The proposed solution to this problem is to apply Machine Learning techniques by using a natural language processor like TF-IDF Vectorizer, classify using Logistic regression and then finally predict who the tweet belongs to.

Highlighting the steps are as follows:

- Read the data and form data frames
- Process the data frames using a python package called TF-IDF Vectorizer and we pull out the n-grams.
- Clean and vectorize input data and create a target vector
- Initialize the model
- Use grid search to find hyperparameters for better results
- Train and fit the optimized model
- Performance evaluation of the model.

Benchmark Model

The benchmark model I will be comparing against is from Tom Wehicle's project (https://github.com/tweichle/Natural-Language-Processing).

Evaluation Metrics

The evaluation metrics, in this case, will be the accuracy score.

Project Design

1. Mine the tweets from both accounts beforehand and extract the required data points into a CSV file
2. Put the data into a pandas dataframe
3. Merge the dataframes into one and pass the new merged data through a python package that does natural language processing called TF-IDF Vectorizer. This will help analyze and rank the n-grams within each tweet.
4. Clean and vectorize the tweet
5. Initialize a model
6. Use gridsearch to find optimal hyperparameters
7. Train and fit the model
8. Evaluate performance by plugging tweets from different people and getting a prediction
9. Present results in a data frame

References

1. Ahmed Besbes, "**Overview and benchmark of traditional and deep learning models in text classification**". https://www.kdnuggets.com/2018/07/overview-benchmark-deep-learning-models-text-classification.html

2. Daniel Jurafsky & James H. Martin. "**Speech and Language Processing. Chapter 3: N-gram Language Models**". https://web.stanford.edu/~jurafsky/slp3/3.pdf

3. Stefan KühnNo "**The Machinery behind Machine Learning – A Benchmark for Linear Regression**". https://blog.codecentric.de/en/2016/01/machinery-linear-regression/

4. Michael Wehicle. “**Natural-Language-Processing: Predicting Political Tweets: Natural Language Processing (NLP) Text Classification Application in Python Using scikit-learn**” https://github.com/tweichle/Natural-Language-Processing