Text Classification Competition: Twitter Sarcasm Detection

CS410 - COURSE PROJECT DOCUMENT

Team – SSW Classifiers

- Saravana Somasundaram (ss129@illinois.edu)
- Shashivrat Pandey (spandey6@illinois.edu)
- Walter Tan (wstan2@illinois.edu)

Table of Contents

1) Introduction to Team: 2

2) Selection of project: 2

3) Project Background: 2

4) Steps followed during implementation: 2

5) Code walkthrough: 3

6) Steps to run the application: 3

7) References: 3

1) Introduction to Team:

We formed a team of three people to work on this project. All of us are passionate about the Data mining concepts were interested on exploring more knowledge around data mining and apply our learning from this course. Our team includes below set of people from fall season of course CS410 from University of Illinois at Urbana-Champaign:

- Saravana Somasundaram (ss129)
- Shashivrat Pandey (spandey6)
- Walter Tan (wstan2)

2) Selection of project:

After discussions, as a team we decided to proceed with Text Classification Competition: Twitter Sarcasm detection project.

Other project topics that we considered are:

- Automatically crawling faculty webpages
- Extracting relevant information from faculty bios

3) Project Background:

In this project there were two sets of data file provided to us to use in our application:

- Training data (train.jsonl)
- Test data(test.jsonl)

We are supposed to build an application to do a prediction for Sarcasm or Not Sarcasm. The data files provided to us are twitter responses. These response texts are in context to some conversation happening in twitter feed. We were supposed to use these two datasets for training our model for this classification. Since this was a classification task, the training file also had the label for each data point as “SARCASM” or “NOT_SARCASM”. Our job was to predict the same thing for all the records present in the testing file. Training file had 5000 labelled dataset and testing set had 1800 dataset. The project required some machine learning task to train the model using the training data and finally predict the outcome for test-data set using that trained classification model. The output of the application will be a text file that capture all the 1800 rows from test-data and SARCASM Vs NOT_SARCASM label. The name of the file is supposed to answer.txt.

4) Steps followed during implementation:

- Analyzed the requirements for the project completion captured under document (CS 410 Project Topics - Google Docs) provided by instructors
- Setup environment to execute the projects
    - Download train.jsonl and test.jsonl data under a folder called data

    - Install all the packages required for project using pip install command
        - Sklearn
        - Pandas
        - Nltk
        - Numpy
- Design a framework to read train.jsonl and test.jsonl files and parse data
- Process the data to remove words such as ‘@USER’ and use the lemmatize function to clean up the data
- Initialize TfidfVectorizer with the appropriate parameters
- Initialize GaussianNB model and train the model using the training data
- Perform predictions of labels using the test data
- Write the results into answer.txt

5) Code walkthrough:

We created and uploaded a video under github account that covers step by step walkthrough of our code implementation.

Below is the URL and Name of the file for code walkthrough video:

- URL: GitHub - ss129/CourseProject
- Name: CourseProject_Demo.mp4

6) Steps to run the application:

Please follow the below steps to run the application and generate the results.

- Clone the below GitHub repository into your local machine - GitHub - ss129/CourseProject
- Install the below packages using pip install command
    - Sklearn
    - Pandas
    - Nltk
    - Numpy
- Execute the python program twitter_sarcasm_classification.py
- The results will be available in the file data/answer.txt

7) References:

a. https://keras.io/examples/nlp/text_classification_from_scratch/
b. https://realpython.com/python-keras-text-classification/#convolutional-neuralnetworks-cnn
c. https://towardsdatascience.com/classification-using-neural-networks-b8e98f3a904f
d. https://towardsdatascience.com/sarcasm-detection-step-towards-sentiment-analysis-84cb013bb6db