Natural Language Processing (2022/2023)

Course Project

ChatGPT sentiment analysis (deadline: 15 May 2023)

Objectives

Practice how to perform text classification using a machine learning classification model and word Embeddings.

ChatGPT has been a major talk in the tech world. There have been several tweets about ChatGPT where users post their opinions towards it. This dataset gathered 219293 tweets for a month, which classifies each tweet into three classes (i.e., positive "good", negative "bad", or neutral).

Example of Dataset (.csv attached with the assignment):

id	Tweet	Opinion
4	ChatGPT about @kunalb11 - English essay writing is going to go for a toss! https://t.co/8t2GKX3Lcl	bad
5	Me: How are you doing?\n\nChatGPT: As a language model trained by OpenAI, I don't have the ability to feel emotions or have experiences. I'm a computer program designed to provide information and answer questions to the best of my ability. Is there something else you'd like to ask? https://t.co/R28Hk0WgAY	good
6	OpenAI ChatGPT: Optimizing Language Models for Dialogue https://t.co/KrH1kX3sZ6 (https://t.co/TbeOPjiX9v)	neutral

The project is consisting of two phases:

Phase 1: is to apply the pre-processing steps which include "Tokenization, stemming, lemmatization, …" "If needed" to clean up and extract the valid final dataset from your data, which will be used to train your model.

You should apply the concept of "word embeddings" to be one of the pre-processing steps while preparing your dataset.

Then you should split your dataset by 80 % - 20 %. The first 80 % will be used as training data for the classifier. While the other 20 % will be used as testing data for your model to be able to identify the "accuracy percentage" of your classifier with the data that have not seen before from the classifier.

Natural Language Processing (2022/2023)

Course Project

Be careful that you should pass to your model the exact expected valid format of data as data is a key factor affecting the accuracy of your model.

Phase 2: Use what you learned over the course to build a classifier that can accept user comments (sentence) and outputs **positive**, **neutral,** or **negative** opinions. You should train your classifier on the data prepared in phase 1. You should produce an average accuracy of at least "85 %" all over the test data that you will use. Randomly divide data to training and testing sets. Note that each set should contain samples of the three types.

Train a classification model to predict the label of the tweet. Train with 2 Models (SVM , CNN, RNN, LSTM..), each model train multiple times to optimize hyper parameters.

Output:

1. Print the accuracy of the model after testing it on the testing set (illustrate results in ipynb).
2. Your program should allow the user to input a new tweet and then predict if it is positive or negative or neural using the trained model (pdf screenshots and illustrate results in ipynb).
3. In the report mention model you used and best hyper parameter for each model, and the accuracy for all trials (graph for each model the number trials per model, mention its hyper parameter).

Follow these steps in your submissions or you will lose your grades (No Excuses):

a) The team should be of 3 members, you can work with any group 😊.
b) You should deliver a folder Containing a subfolder for the training data, a folder for testing data and a folder containing your python file(s) or ipynb.
c) Name the folder with your IDs separated by underscores without anyspaces_your group number. (Example: 2011111_20145666_20135555_S7)
d) Compress this folder in a .zip file with the same folder name and upload it.
e) Only one member from the group should upload the project artifacts.

Good Luck 😊