COMP5423 NATURAL LANGUAGE PROCESSING

Lab1 Homework: Emotion Classification

**Due date**: 11:59pm, 27/2/2022 (Sunday)

**Group Size**: Individual

**Project Objective**

The goal of this project is to learn how to process the text data and classify the emotion of a text input. For example, “I feels so lame” reflects a “sadness” emotion. We provide an emotional classification dataset(data.zip). In the Kaggle dataset, each sentence is labelled with one emotion (joy, fear, love, anger, sadness, surprise). The dataset includes training, validation, and testing set, respectively. The training and validation sets are labeled, and the testing data is not labeled. Their basic statistics are as below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | #train | #validation | #test | marks |
| Kaggle | 16000 | 2000 | 2000 | Labelled with emotions |

**Requirements**

1. Process the text input.
2. Design your method to classify the emotions.
3. Compare and analyze the results (the classification accuracy on the test set).
4. If possible, design an UI system (like via the Web programing or a simple command environment) for your method. Specifically, for each input sentence, when you classify its emotion, we hope your UI system can choose a corresponding emoji (for the joy emotion, the corresponding emoji may be a smile face) to show the predicted emotion.

**What to Hand in and Grading Scheme**

Pack all files in one zipped file with the clear name for each file and submit I to Blackboard.

1. The Source Code of Your Program (including UI system)

(1) A readme file describes the structure of your program and how to run it.

(2) Add annotations in the front of each (Python) file to specify its usage.

(3) Clear comments of your code.

1. Report
2. How you process the text input.
3. How you classify the emotions.
4. The results on the test set.
5. The UI system. Describe its function and its workflow. Some screenshots are necessary. (online systems are not necessary)

3. Predicted result on the test set

Submit a txt file named test\_prediction.txt, where each line is the predicted emotion of the corresponding text in the test\_data.txt. We provide a submission txt file that you can refer to (test\_prediction\_sample.txt).

**Remarks\*:**

1. A good accuracy on the test set **don’t guarantee** a high score of this project, your design and a clear description of your design are also valuable.
2. You can implement this conversational system with the assistance of **external knowledge**
3. You are strongly recommended to implement the system in **Python**.

**Reference:**

1. Related Work:

* Affect Analysis Model: novel rule-based approach to affect sensing from text
* [Finding People with Emotional Distress in Online Social Media: A Design Combining Machine Learning and Rule-Based Classification.](http://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=02767783&AN=143513878&h=nSSISVP3YoiA%2FaDFFXJuABhiIT9oWkM31B5xxvBTMQg%2BFBuffgE2u0cPzQMwkGYxZIaJp7qKsBYQEEABXugKXg%3D%3D&crl=c)
* MojiTalk: Generating Emotional Responses at Scale
* Emotional Chatting Machine: Emotional Conversation Generation with Internal and External Memory

1. Free GPU resources:

* Baidu PaddlePaddle: [courese registeration](https://aistudio.baidu.com/aistudio/course/introduce/1022) [tutorial](https://aistudio.baidu.com/aistudio/course/introduce/1022) [usage](https://mp.weixin.qq.com/s/nKgjgPUYb7uoCRhak0rshg)
* Google Colab: [tutorial](https://medium.com/deep-learning-turkey/google-colab-free-gpu-tutorial-e113627b9f5d) [usage](https://colab.research.google.com/github/tensorflow/tpu/blob/master/tools/colab/bert_finetuning_with_cloud_tpus.ipynb)
* Kaggle Kernel: [tutorial](https://www.kaggle.com/dansbecker/running-kaggle-kernels-with-a-gpu) [usage](https://zhuanlan.zhihu.com/p/60912138)
* FolydHub: [usage](https://zhuanlan.zhihu.com/p/29874845)