Foundations of Natural Language Processing

Peking University, 2025 Spring

Assignment 2: Due on Wednesday, April 30, 2025 by 11:59 PM

Task Description

In this assignment, you will implement and train two models for text classification: a **log-linear model** and a **BERT model**. For the log-linear model, you will need to extract features from document `d`, including but not limited to **unigrams**, **bigrams**, or **tf-idf** representations. Additional feature engineering is encouraged. For the BERT model, to ensure consistent comparisons, we will exclusively use google-bert/bert-base-uncased.

You may utilize any relevant toolkits for implementation (e.g., **scikit-learn**, **HuggingFace Transformers**, etc.). Model performance should be evaluated using **Accuracy**, **Macro-F1**, and **Micro-F1** metrics.

The following datasets will be used:

1. **20-Newsgroups**
   This dataset contains approximately 20,000 newsgroup documents evenly distributed across 20 categories.
2. You should load the dataset using the huggingface datasets library:

```python
from datasets import load_dataset
dataset = load_dataset('SetFit/20_newsgroups')
```

Note: If you experience network connectivity issues, you may access the dataset via the Hugging Face mirror site: https://hf-mirror.com/

2. **Hallmarks of Cancer Corpus (HoC)**
   The HoC dataset comprises expert-annotated publication abstracts classified according to a 11-category taxonomy. We have preprocessed the original multi-label dataset to include only single-label instances with balanced class distribution. The dataset files are located at:
   - `./data/HoC/train.parquet` and `./data/HoC/test.parquet`

Submission Requirements

Your submission must be a `.zip` file include the following components:

1. **Code**
2. **Report** in pdf format, which should contain at minimum:
   1. **Implementation details** for both log-linear and BERT models (you can include relevant code excerpts, but be **concise**)
   2. **Performance results** on training and test sets of both datasets
   3. **Analysis** on the comparison of classification results among different models, the impact of dataset characteristics on the results, and so on.

3. A `results.json` file documenting your models' performance on both training and test sets. Maintain the following structure:

```
{
    "20newsgroups": {
        "train": {
            "accuracy": 0.9999,
            "macro_f1": 0.9999,
            "micro_f1": 0.9999
        },
        "test": {
            "accuracy": 0.9999,
            "macro_f1": 0.9999,
            "micro_f1": 0.9999
        }
    },
    "HoC": {
        "train": {
            ...
        },
        "test": {
            ...
        }
    }
}
```

Important Guidelines

1. **Shell Script Requirement**

   You must submit an executable shell script named `run.sh`. This script will be used to execute your program during grading. Failure to provide a functional script will result in a **maximum score of 50%**.

   Notes:

   - The script, source code, and data files reside in the same directory
   - Use **relative paths** exclusively in both your script and source code

2. **Late Submission Policy**

   Late submissions will incur a **5% penalty** per day, applied to the final grade. **No submissions will be accepted after May 7, 2025** (one week after the due date). Submission times are determined by your final upload timestamp on https://course.pku.edu.cn/. Resubmissions after the deadline deadline should only be attempted if the improvements outweigh the late penalty. Multiple resubmissions are permitted, but ensure all required files are included in each submission. Deadline enforcement is strict and based on the timestamp of your last submission. Penalties will be rounded to the nearest integer percentage point.

3. **Contact Information**: For any inquiries, please contact the TAs:

   - linjiuheng@stu.pku.edu.cn (林九衡)
   - luokangcheng@stu.pku.edu.cn (罗康诚)