4NL3_Homework_4

Muhammad Muneeb Hassan

April 2025

1 Dataset

I used the sms spam detection dataset available at Hugging face website. The link to the dataset can be found at the homework 4 pdf. The dataset, only had a train section. Therefore I had to break the dataset into **Train** and **Test** dataset with 80-20 ratio.

Table 1: Training and Test Data Splits

Split	**Size**	**Class Distribution**
Training Set	4,459	78% Class 0 (not spam) 22% Class 1 (spam)
Test Set	1,115	77.3% Class 0 (not spam) 22.7% Class 1 (spam)

2 Fine-tuned models

The models I chose were the BERT and the RoBERTa model.

2.1 Steps

I first had to load the model. After this, I had to define the tokenizer which tokenizes the text based on the model I used. I then had to setup the training trainer which included the parameters such as epochs and learning rate. The trainer was then fine-tuned on my dataset and then used to perform evaluations on the dataset.

2.2 BERT

- **Parameters:** 110M
- **Pre-trained data:** BooksCorpus (800M words) + English Wikipedia (2.5B words)
- **Compute:** Trained on 16 TPUs for 4 days

2.3 RoBERTa

- **Parameters:** 125M
- **Pre-trained data:** BooksCorpus, Wikipedia
- **Compute:** Trained on 1024 V100 GPUs for 1 day (batch size: 8K)

3 Zero-shot classification

The models that I used were typeform/distilbert-base-uncased-mnli and the GPT-2 model.
Typeform/distilbert-base-uncased-mnli model has 66M parameters and pre-trained similar to BERT. The GPT-2 model had close to 200M parameters and was pre-trained on web-text. I prompted the model by trying out a few different prompts to see which ones gave the best probability of classifiers with respect to the text. Following were a couple of the prompts I tried.

- Classify the following text into one of these categories: spam, ham. Text: "text" Answer:
- Is this text offensive? Answer yes or no: "text"

4 Baselines

I implemented a logistic regression classifier using Bow approach on my dataset. I split up the dataset into train and test set, the used the classification to classify the test data. Similarly, for majority baseline, I created a seperated predicted labels, all consisting of 0's (which was the majority label) and then compared it to the test set label to compute the accuracy. For random baseline, I created predicted labels with random values (0 or 1) and then compared it with the test set label.

5 Results

article booktabs

The results show that fine-tuned transformer models (BERT/RoBERTa) achieve near-perfect accuracy (99+) for spam detection, while simpler logistic regression with BoW performs surprisingly well (98). Zero-shot models like GPT-2 and DistilBERT-MNLI perform poorly (16-39). The high majority baseline and random baseline perform moderadely well as well since there were only 2 classes. Eventhough, fine-tuned models performed the best, I believe simpler models like logistic regression would be best for these kinds of tasks since there are only 2 classifiers, the dataset is small, and would require much less computation power.

Table 2: Model Performance on SMS Spam Detection

Model	**Accuracy (%)**
RoBERTa	99.28
BERT	99.19
Logistic Regression (BoW)	98.00
Majority Baseline	78.00
GPT-2	38.70
typeform/distilbert-base-uncased-mnli	16.70
Random Baseline	50.00

6 reflection

One of the main things I was looking forward to ever since the start of the course was the fine-tuned transformers like BERT and how these technologies work. This project helped me really understand the concepts behind these technologies. Initially, I didnot knew that I had to use a gpu for processing. Therefore, I initially continued with the cpu device and ran into some computation problems. However, soon I fixed it and rand the programs on gpu. The zero-shot classification was definitely the most challenging part for me because of the lack of resources surrounding it.

7 Citations

I made use of chat gpt and deep seek throughout this assignment. However, majority of the code is written by myself and understood.