Sentiment Analysis Assignment Report

CSE674: Text Analytics - Fall 2024

Instructor: Sajjad Haider

Contributors:

Saad Ullah (29416)
Farjad Ahmed (29405)
Muhammad Ghulam Abbas (29417)

Contents

Credits

The successful completion of this project would not have been possible without the contributions of all team members. The tasks were divided as follows:

- **Saad Ullah**: Saad was primarily responsible for fine-tuning the DistilBERT model, conducting extensive benchmarking, and analyzing the results. He led the experiment configurations, including the testing and evaluation of different LoRA setups for DistilBERT. Additionally, Saad contributed to the overall report writing, particularly in the discussion of PLM results and performance comparison.
- **Farjad Ahmed**: Farjad took the lead in implementing the classical machine learning (ML) models. He preprocessed the text data, handled tokenization, stopword removal, stemming/lemmatization, and POS tagging, and also performed the vectorization using TF-IDF and CountVectorizer. Farjad was instrumental in training and evaluating models like Naive Bayes, Logistic Regression, k-NN, and Random Forest, and contributed significantly to the results section of the report.
- **Muhammad Ghulam Abbas**: Ghulam was responsible for the fine-tuning of the RoBERTa model, experimenting with various LoRA configurations, and analyzing the results. He also supported in the comparative analysis of DistilBERT and RoBERTa, and contributed to the insights and discussion on the trade-offs between the classical ML and PLM approaches. Ghulam also assisted in structuring and finalizing the report.

Each member contributed effectively to the completion of the assignment, ensuring that the methodology was well-executed and the results were thoroughly analyzed and documented.

1. Introduction

The objective of this assignment is to compare classical machine learning (ML) methods and pre-trained language models (PLMs) for sentiment analysis. Classical ML involves preprocessing text data, vectorization, and training models such as Naive Bayes, Logistic Regression, k-NN, and Random Forest. In contrast, PLMs such as DistilBERT and RoBERTa are fine-tuned with Low-Rank Adaptation (LoRA) layers for efficient parameter optimization. Performance is evaluated based on accuracy, precision, recall, and F1 score.

2. Methodology

2.1 Classical Machine Learning Approaches

Preprocessing:

- **Tokenization:** Split text into words or tokens.
- **Stopword Removal:** Eliminated non-informative words.
- **Stemming/Lemmatization:** Reduced words to their base forms.
- **POS Tagging:** Identified parts of speech for potential feature engineering.

Vectorization:

- Used TF-IDF for feature extraction with varying n-gram ranges.
- Experimented with CountVectorizer for baseline feature representation.

Models:

- Trained Naive Bayes, Logistic Regression, k-NN, and Random Forest using preprocessed data.

2.2 Fine-Tuning Pre-trained Language Models (PLMs)

Models Used:

- DistilBERT
- RoBERTa

LoRA Configurations:

- Experimented with rank, alpha, batch size, and dropout.
- Fine-tuned on sentiment-labeled data with three different configurations.

Evaluation Metrics:

- Accuracy, Precision, Recall, and F1-Score.

3. Experiments and Results

3.1 Classical Machine Learning Results

Model	Accuracy
Naive Bayes	87.98%
Logistic Regression	89.27%
k-NN	79.45%
Random Forest	85.49%

3.2 Fine-Tuning DistilBERT

Summary of Experiments

Config	Learning Rate	Epochs	Batch Size	LoRA Rank	LoRA Alpha	LoRA Dropout	Accuracy
Config_1	0.00005	1	2	4	8	0.10	87.39%
Config_2	0.00003	1	2	8	16	0.20	87.03%
Config_3	0.00001	2	2	2	4	0.05	86.76%

Best Configuration: Config_1

Testing Results (Config_1)

- **Accuracy:** 87.15%
- **Precision:**
  - Class 0: 78.0%
  - Class 1: 89.1%
- **Recall:**
  - Class 0: 89.6%
  - Class 1: 84.9%
- **F1-Score:**
  - Class 0: 83.5%
  - Class 1: 86.9%
- **Macro Average:**
  - Precision: 83.6%
  - Recall: 87.3%
  - F1-Score: 85.2%

3.3 Fine-Tuning RoBERTa

Summary of Experiments

Config	Learning Rate	Epochs	Batch Size	LoRA Rank	LoRA Alpha	LoRA Dropout	Accuracy
Config_1	0.00005	1	2	4	8	0.10	78.00%
Config_2	0.00003	1	2	8	16	0.20	77.90%
Config_3	0.00001	2	2	2	4	77.65%	

Best Configuration: Config_1

Testing Results (Config_1)

- **Accuracy:** 78.00%
- **Eval Loss:** 0.6939

3.4 Benchmarking Results

The benchmark model "distilbert-base-uncased-finetuned-sst-2-english" was evaluated, achieving the following results:

Class	Precision	Recall	F1-Score	Support
0	0.93	0.86	0.89	966
1	0.88	0.94	0.91	1034

Metric	Value
Eval Loss	0.2563
Eval Runtime	20.07s
Eval Samples/s	99.65
Eval Steps/s	6.23
Epoch	3.0

Overall Performance:

- **Accuracy:** 90%
- **Macro Average:**
  - Precision: 0.90
  - Recall: 0.90
  - F1-Score: 0.90
- **Weighted Average:**
  - Precision: 0.90
  - Recall: 0.90
  - F1-Score: 0.90

4. Performance Comparison

Model	Accuracy	Training Time	Resource Requirements
Naive Bayes	87.98%	Low	Minimal
Logistic Regression	89.27%	Moderate	Minimal
k-NN	79.45%	Moderate	Moderate
Random Forest	85.49%	High	Moderate
DistilBERT	87.15%	Very High	GPU-intensive
RoBERTa	78.00%	Very High	GPU-intensive
Benchmark Model	90.00%	Very High	GPU-intensive

---

5. Insights and Discussion

Classical ML Models:

- Logistic Regression achieved the highest accuracy of 89.27%, outperforming other classical models.
- Naive Bayes performed well despite being less computationally intensive.
- k-NN struggled due to the high dimensionality of text data.

DistilBERT:

- Fine-tuning with LoRA achieved comparable performance to classical ML with Config_1 (87.15%).
- Config_1 outperformed other configurations due to optimized LoRA rank and alpha values.

RoBERTa:

- Underperformed compared to both DistilBERT and classical ML models, possibly due to insufficient fine-tuning epochs and smaller batch size.

Benchmark Model Comparison:

- The benchmark model achieved the highest accuracy (90%), surpassing all other models, including fine-tuned DistilBERT and Logistic Regression.
- Its superior performance highlights the effectiveness of pre-trained models fine-tuned for specific tasks.
- However, the benchmark model also required significantly higher computational resources and training time.

Training Time and Resources:

- PLMs required significantly more resources (GPU) and training time compared to classical ML models.

Strengths and Weaknesses:

- Classical ML models are lightweight and effective for small datasets.
- PLMs excel in capturing contextual nuances but require high computational power.

6. Conclusion

- **Best Classical Model:** Logistic Regression with 89.27% accuracy.
- **Best PLM:** DistilBERT with Config_1 achieving 87.15% accuracy.
- **Overall Best Model:** Benchmark model with 90% accuracy.
- Classical ML models are more resource-efficient but lack the contextual depth of PLMs.
- PLMs offer strong performance on larger datasets or nuanced tasks, making them preferable when resources are available.

7. References

- Vaswani et al., “Attention is All You Need”, 2017.
- Raffel et al., “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer”, 2019.
- Additional course materials and research papers provided in CSE674.