NLU course projects

Ettore Saggiorato (247178)

University of Trento

ettore.saggiorato@studenti.unitn.it

1. Introduction

This report presents the implementation and evaluation of a model for extracting aspect terms in the Aspect-Based Sentiment Analysis (ABSA) task by fine-tuning BERT [1], as the third assignment for the course of Natural Language Understanding at the University of Trento. The used dataset is the `laptop14` partition from the SemEval-2014 Task 4 dataset[2], which is a well-known benchmark for ABSA. The employed evaluation metrics are precision, recall, and F1 score, based on the official SemEval evaluation script [3][4].

The goal of this assignment is to fine-tune BERT for the specific task of aspect term extraction, which is conceptually similar to the slot filling task encountered in the second assignment of the course. This similarity extends to the model architecture and training procedures.

2. Implementation Details

2.1. Dataset Preprocessing

The dataset consists of text files where each line represents an annotated sentence. An example of an entry is as follows:

```
Boot time is super fast, around
anywhere from 35 seconds to 1
minute.Boot=T-POS time=T-POS
is=O super=O fast=O ,=O around=O
anywhere=O from=O 35=O seconds=O
to=O 1=O minute=O .=O
```

Each sentence is processed individually, with each token and its corresponding tag mapped into the BIESO format. This mapping is essential for compatibility with the evaluation function, which requires this tagging scheme. After preprocessing, the data is split into training, testing, and evaluation sets, and corresponding data loaders are configured. Batches are dynamically created with minimal padding to optimize memory usage and reduce the presence of non-informative padding tokens, which can hinder model learning.

As in the second assignment, the used model and tokenizer is `bert-base-uncased`. Two dictionaries (`slots2id` and `id2slots`) are employed to map aspect term tags to unique IDs and vice versa. The input sentences are tokenized using BERT's tokenizer to ensure compatibility with the model.

2.2. Fine-Tuning

This process follows guidelines from Bert's paper [1]. Training is conducted over six epochs, a reduction from the ten epochs typically used in JointBERT[5][6], with each experiment repeated five times to average the results, thereby ensuring consistency.

To improve training stability, gradient clipping is employed to mitigate the risk of exploding gradients. Bert's learning rate scheduler is also utilized to dynamically adjust the learning rate throughout the training process, aiding in convergence. Early stopping is not implemented in this experiment.

3. Results

The results are presented in Table 1. The results indicates that gradient clipping significantly enhances model performance, getting better precision and F1, while the learning rate scheduler improves only on recall.

Model	Clip.	Sch.	F1	Precision	Recall
simple_32	F	F	78.39%	78.90%	77.89%
grad_32	T	F	**80.11%**	**80.93%**	79.46%
sch_32	F	T	78.61%	77.79%	79.46%
sch_grad_32	T	T	78.76%	77.41%	**80.17%**

Table 1: *Experiment Results* [1]

4. References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[2] X. Li, "E2e-tbsa," https://github.com/lixin4ever/E2E-TBSA/tree/master/data.

[3] ——, "Evaluation script for semeval-2014 task 4," https://github.com/lixin4ever/E2E-TBSA/blob/master/evals.py.

[4] SemEval 2014 Organizers, "Semeval-2014 task 4: Aspect based sentiment analysis - data and tools," 2014. [Online]. Available: https://alt.qcri.org/semeval2014/task4/index.php?id=data-and-tools

[5] P. Jangwon, "Jointbert," https://github.com/monologg/JointBERT/tree/master.

[6] C.-k. Lee and S. Ting, "Robust multi-task learning for natural language understanding," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. ACL, 2019, pp. 1247–1254.