NLU project exercise lab: 11

Carlo Marotta (231590)

University of Trento
carlo.marotta@studenti.unitn.it

1. Introduction

Sentiment analysis, a crucial task involving the extraction and evaluation of emotions and opinions expressed in text, holds significant importance within the field of natural language processing. This paper offers a comprehensive examination of sentiment analysis, delving into various dimensions, including the detection of **subjectivity**, prediction of **polarity**, and **aspect-based sentiment analysis**. Employing state-of-the-art neural network models and pretrained language models, our aim is to offer a comprehensive insight into the world of sentiment analysis tasks and their intricate interplay.

2. Implementation details

In the first part of our exercise, we tackled the creation of a pipeline model for **subjectivity** and **polarity detection tasks**. This pipeline consists of three main functions: `load_subjectivity_data`, `load_polarity_data` and `objective_removal`. The first two functions preprocess the working datasets (*Subjectivity* and *Movie Reviews* respectively) through a series of text cleaning operations, such as removing stop words, punctuation and lemmatizing words. The result obtained is a list of sentences associated with their respective subjectivity or polarity labels.
In the case of polarity with the removal of objective sentences, the `objective_removal` function takes sentences and labels from the polarity dataset and applies the subjectivity model to remove objective sentences. The extraction of subjective sentences can only occur when a pre-trained model for subjectivity task is available. This process is a pivotal step in the pipeline for polarity detection task and aims to obtain a more focused dataset containing only subjective sentences on which to evaluate performance improvements.
In the second part of the exercise, we focused on **aspect-based sentiment analysis** using the *Laptop partition* dataset from SemEval2014 task 4. We extracted aspect terms from the texts and then performed polarity detection for these terms. To do this, we loaded the dataset and preprocessed it to obtain a list of aspect terms and their respective polarities for subsequent analysis with our model. The crucial functions for the success of this part are `get_ot` for aspect terms extraction and `get_ts` for polarity detection, based on iterative calls to the built-in function `convert_polarity_in_ts`. This function allowed us to extract from the dataset a list of sentiments (*POS* for positive, *NEG* for negative, *NEU* for neutral or *O* for null) associated with their structural position in the text (*"S"* for singleton,*"B"* for the beginning of a sequence of equal sentiments, *"I"* for inside a sequence with a length greater than 2, and *"E"* for the end of a sequence).
In all the models implemented for this exercise, we used **BERT**, as the pre-trained model, and the AdamW optimizer. To optimize model performance, we primarily focused on tuning `LEARNING_RATE` parameters and batch sizes.

3. Results

For evaluating the performance of our models across all exercises, we designed a central function `train`. Within this function, we initialized the models, defined the loss functions and set up the optimizer, and it is structured to seamlessly integrate both the training and testing phases by invoking `train_loop` and `test_loop`, respectively. The training function is responsible for tasks such as loss computation, parameter updates using the AdamW optimizer, and iteratively improving the model's performance. Concurrently, the testing phase handled the calculation of accuracy to assess the model's effectiveness. To identify the most optimal model, we implemented an early stopping mechanism, configured with a parameter denoted as `PATIENCE`. This early stopping mechanism not only assisted us in identifying the top-performing model but also served as a safeguard against overfitting to the training data by halting the training iteration as soon as the patience level reached zero.
For the aspect-based sentiment analysis task, the ultimate evaluation criterion isn't the accuracy of the pre-trained model. Instead, this model serves as a foundational tool for the proper assessment of performance in two important areas: the extraction of aspect terms and the detection of their respective polarities. The central function in this context is `evaluate`, which combines **Opinion Target Extraction** (**OTE**) and **Targeted Sentiment** (**TS**) assessments, by calling both `evaluate_ote` and `evaluate_ts`. The former evaluates OTE performance, while the latter assesses TS performance, both based on *F1-score*, *precision* and *recall* calculations. This entire evaluation process is encapsulated within the *evals.py* file, which ensures a comprehensive evaluation of the model's performance on aspect term extraction and polarity detection tasks.

3.1. Part 1

For the subjectivity task, we conducted several experiments to assess the best model performance by tuning the `LEARNING_RATE` parameter. However, in the case of the two models for the polarity task, conducting a substantial number of experiments to evaluate optimal performance was challenging due to significant computational resource limitations, even when utilizing a virtual GPU machine. Hence, we opted to utilize the same hyperparameters that were selected for the best-performing model in the subjectivity task. We can indeed confirm that there has been an improvement in final accuracy (not the mean) when using the dataset without objective sentences as opposed to the performance achieved with the entire dataset.

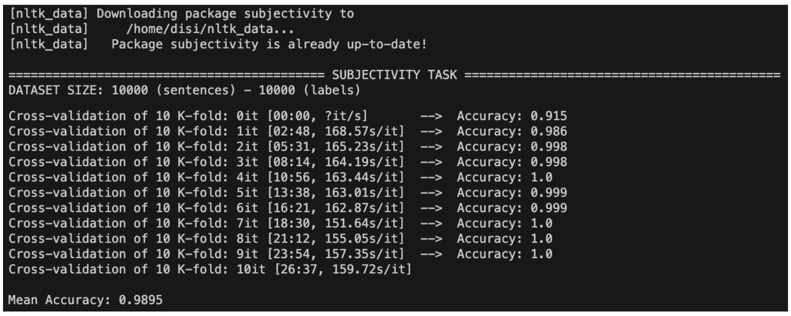
3.2. Part 2

In this scenario, constrained by the large dataset and limited GPU availability, we were able to conduct only a limited number of experiments. Nevertheless, we achieved promising results, ensuring meaningful performance for the final evaluation of Aspect Terms Extraction and Polarity Detection tasks.

4. Showing results

Table 1: *Part 1 and Part 2*

TASK	BATCH_SIZE	LEARNING_RATE	**Accuracy**	**F1**	**Precision**	**Recall**
Subjectivity	64/64	0.00005	0.9895	-	-	-
Polarity	64/64	0.00005	0.9831	-	-	-
Subjectivity & Polarity	64/64	0.00005	0.9788	-	-	-
Aspect Term Extraction	1024/1024	0.00005	0.9293	0.97453	0.95142	0.99891
Polarity Detection	1024/1024	0.00005	0.9293	0.93535	0.92197	0.94922



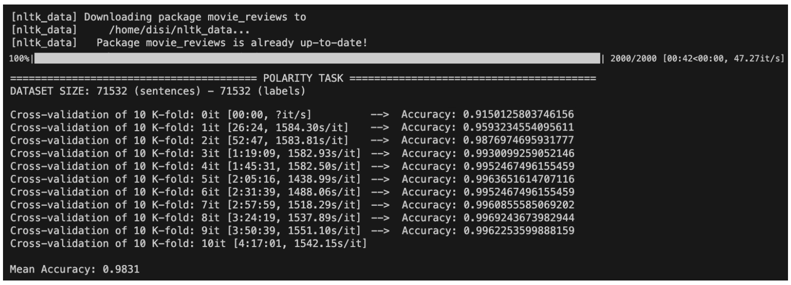
```
[nltk_data] Downloading package subjectivity to
[nltk_data]     /home/disi/nltk_data...
[nltk_data]   Package subjectivity is already up-to-date!

=========================================== SUBJECTIVITY TASK ===========================================
DATASET SIZE: 10000 (sentences) - 10000 (labels)

Cross-validation of 10 K-fold: 0it [00:00, ?it/s]       -->  Accuracy: 0.915
Cross-validation of 10 K-fold: 1it [02:48, 168.57s/it]  -->  Accuracy: 0.986
Cross-validation of 10 K-fold: 2it [05:31, 165.23s/it]  -->  Accuracy: 0.998
Cross-validation of 10 K-fold: 3it [08:14, 164.19s/it]  -->  Accuracy: 0.998
Cross-validation of 10 K-fold: 4it [10:56, 163.44s/it]  -->  Accuracy: 1.0
Cross-validation of 10 K-fold: 5it [13:38, 163.01s/it]  -->  Accuracy: 0.999
Cross-validation of 10 K-fold: 6it [16:21, 162.87s/it]  -->  Accuracy: 0.999
Cross-validation of 10 K-fold: 7it [18:30, 151.64s/it]  -->  Accuracy: 1.0
Cross-validation of 10 K-fold: 8it [21:12, 155.05s/it]  -->  Accuracy: 1.0
Cross-validation of 10 K-fold: 9it [23:54, 157.35s/it]  -->  Accuracy: 1.0
Cross-validation of 10 K-fold: 10it [26:37, 159.72s/it]

Mean Accuracy: 0.9895
```

Figure 1: *Subjectivity task*



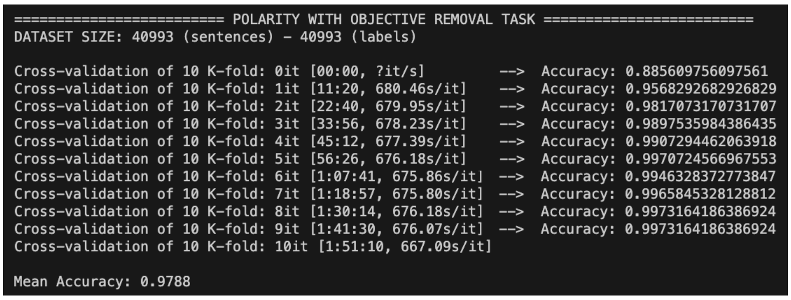
```
[nltk_data] Downloading package movie_reviews to
[nltk_data]     /home/disi/nltk_data...
[nltk_data]   Package movie_reviews is already up-to-date!
100%|██████████████████████████████████████████████████████████████| 2000/2000 [00:42<00:00, 47.27it/s]

======================================== POLARITY TASK ========================================
DATASET SIZE: 71532 (sentences) - 71532 (labels)

Cross-validation of 10 K-fold: 0it [00:00, ?it/s]          -->  Accuracy: 0.9150125803746156
Cross-validation of 10 K-fold: 1it [26:24, 1584.30s/it]    -->  Accuracy: 0.9593234554095611
Cross-validation of 10 K-fold: 2it [52:47, 1583.81s/it]    -->  Accuracv: 0.9876974695931777
Cross-validation of 10 K-fold: 3it [1:19:09, 1582.93s/it]  -->  Accuracy: 0.9930099259052146
Cross-validation of 10 K-fold: 4it [1:45:31, 1582.50s/it]  -->  Accuracy: 0.9952467496155459
Cross-validation of 10 K-fold: 5it [2:05:16, 1438.99s/it]  -->  Accuracy: 0.9963651614707116
Cross-validation of 10 K-fold: 6it [2:31:39, 1488.06s/it]  -->  Accuracy: 0.9952467496155459
Cross-validation of 10 K-fold: 7it [2:57:59, 1518.29s/it]  -->  Accuracy: 0.9960855585069202
Cross-validation of 10 K-fold: 8it [3:24:19, 1537.89s/it]  -->  Accuracy: 0.9969243673982944
Cross-validation of 10 K-fold: 9it [3:50:39, 1551.10s/it]  -->  Accuracy: 0.9962253599888159
Cross-validation of 10 K-fold: 10it [4:17:01, 1542.15s/it]

Mean Accuracy: 0.9831
```

Figure 2: *Polarity task*

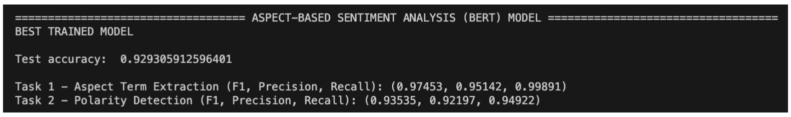


```
========================= POLARITY WITH OBJECTIVE REMOVAL TASK =========================
DATASET SIZE: 40993 (sentences) - 40993 (labels)

Cross-validation of 10 K-fold: 0it [00:00, ?it/s]         -->  Accuracy: 0.885609756097561
Cross-validation of 10 K-fold: 1it [11:20, 680.46s/it]    -->  Accuracy: 0.9568292682926829
Cross-validation of 10 K-fold: 2it [22:40, 679.95s/it]    -->  Accuracy: 0.9817073170731707
Cross-validation of 10 K-fold: 3it [33:56, 678.23s/it]    -->  Accuracy: 0.9897535984386435
Cross-validation of 10 K-fold: 4it [45:12, 677.39s/it]    -->  Accuracy: 0.9907294462063918
Cross-validation of 10 K-fold: 5it [56:26, 676.18s/it]    -->  Accuracy: 0.9970724566967553
Cross-validation of 10 K-fold: 6it [1:07:41, 675.86s/it]  -->  Accuracy: 0.9946328372773847
Cross-validation of 10 K-fold: 7it [1:18:57, 675.80s/it]  -->  Accuracy: 0.9965845328128812
Cross-validation of 10 K-fold: 8it [1:30:14, 676.18s/it]  -->  Accuracy: 0.9973164186386924
Cross-validation of 10 K-fold: 9it [1:41:30, 676.07s/it]  -->  Accuracy: 0.9973164186386924
Cross-validation of 10 K-fold: 10it [1:51:10, 667.09s/it]

Mean Accuracy: 0.9788
```

Figure 3: *Polarity task with objective sentences removal*



```
=================================== ASPECT-BASED SENTIMENT ANALYSIS (BERT) MODEL ===================================
BEST TRAINED MODEL

Test accuracy:  0.929305912596401

Task 1 - Aspect Term Extraction (F1, Precision, Recall): (0.97453, 0.95142, 0.99891)
Task 2 - Polarity Detection (F1, Precision, Recall): (0.93535, 0.92197, 0.94922)
```

Figure 4: *Aspect Terms Extraction and Polarity Detection tasks*