Homework 2 Written Report

NLP 2021 Course - Prof. Roberto Navigli

Andrea Trianni - 1806198
Master Degree in Computer Science
Sapienza University of Rome
trianni.1806198@studenti.uniroma1.it

1 Introduction

The topic of this homework is Aspect Based Sentiment Analysis, that consist in identify aspect target words in a sentence and also predict their corresponding sentiment. The homework pratically is made of 4 different subtask, namely **A** (*Aspect term extraction*), **B** (*Aspect sentiment classification*), **C** (*Category extraction*), **D** (*Category sentiment classification*). These tasks has to be performed on two dataset domains: restaurant and laptop text reviews.
In this report, i am going to illustrate my approaches to solve individually each one of this task, and i will introduce also a final model for a unified End-To-End solution to the problem.

2 Data Visualization

Both the datasets and the task are widely descript here (Pontiki et al., 2014), anyway, before proceeding, it is a good practice to view and analyze the data we are going to use, to build models that can fit well the problem.
As shown in figure [2], the length of the sentences is distributed like a gaussian, analyzing it, i decide to fix the max size of the sentence to 80 token. This admit to fit most of the text samples and to not heavily pad some other sentences, that can add undesired noise in some cases.
A problem of both datasets, for what concerns sentiment task B and D, is class unbalancing, as shown in image [5] [6]. The labels of the datasets are very unbalanced both in training and dev set. This is not fine, since it can affect the models performances, moreover considering that the metric of evaluation will be macro-f1 score. For these reasons i decided to use weighted loss, afterwards i have discarded the earlier idea to delete "conflict" label from the dataset. Class unbalancing is still present in task C [7], but it is not so heavy as for B and D, so in this case i don't care about fighting the problem.
Meanwhile models for task C and D will be trained only on restaurant reviews, for task A and B the models will be trained both on laptop and restaurant dataset, that will be merged and shuffled together. This will enable the model to have not only more sample to learn, but also to generalize better, since we are performing a joint-domain learning. The two datasets have exactly the same size, so i am not injecting any bias in doing this operation [1].
Continuing to discuss about task A, in figure [3], we can notice that most of the target consist of only one token, this is important because the length can affect the performance of different encoding strategy for the target, that will be deeply discuss in section 4.1. Another important info that came up visualizing the data is that in restaurant dataset there are less different target words, that are consequently more frequent, respect laptop dataset. This means that the models can suffer when predicting aspect from laptop sentences.
All this information is fundamental for the choices i've made in the design of the models.

3 Preprocessing stage

For all the models of this homework, i am going to use pretrained BERT-base-cased (Devlin et al., 2019), so i will use standard bert vocabulary and standard bert word-piece tokenizer. I have chosen the cased version because uppercase or lowercase words can express different level of sentiment, and in this way i can get the best performance from my models.
BERT provides contextualized word embedding that can offer a more accurate representation respect standard word embedding like Glove, that i previously used in hw1. This is a crucial point since the sentiment analysis is aspect based and a

sentence can have more aspect and different sentiments, so contextual information are very useful to discriminate, understand words relations, and help the model in training phase.
For each task, a different data processing pipeline has been performed and a different encoding strategy has been chosen. These design choices will be discussed when i will introduce my models for each task, in the next section.

4 Models

4.1 Task A

I solved task A as a sequence labeling problem, in a similar way respect other well known task of NLP, like named entity recognition. So, given a sentence $\{w1, w2, .., wn\}$, i have to assign to each word w a label. For the labels, i tried to use 2 different encoding strategy: the famous BIO $\{Begin, Inside, Outside\}$ and IO $\{Inside, Outside\}$, for all the reasoning i have done in the data visualization section, about aspect length. So the goal is to assign to each token a label in the set. To reach this goal, my model consist of fine-tuned BERT that feed, with the last output layer, a simple linear classifier with dropout. Final probabilities are extracted with a softmax function. So the model outputs a sequence of labels $\{l1, l2, .., ln\}$, that will be decoded to string to feed model b for sentiment classification. The conversion from token tagging to string is quite simple and it works in the way you can imagine. During the coding stage i tried also a solution with a final linear-CRF, that replace the softmax function. Conditional Random Field is espected to be a more powerful architecture, since it doesn't classify each token independently, but it try to find the best sequence of tags. I managed to get performance slightly better than the delivered model with softmax ($\simeq +2\%$), but then i discarded that model (also from code), since CRF is not a topic that has been covered during the course.

4.2 Task B

Task B is more than a standard sentiment classification problem, because i have to take into account that a sentence can have more aspect and of course each aspect can have different sentiment, so we have to assign, given the whole sentence, a sentiment to each aspect target.
As for the other tasks i decided to use bert, so the architecture i have designed is slightly inspired by the paper (Sun et al., 2019). This because i need a clever way to exploit transfer learning, and a good strategy is to solve this task as a sentence pair classification problem.
The first sentence is, of course, the review, and the second one is simply the aspect term. Bert accept in input the pair of token sequences that are separated by [SEP] token, meanwhile the output is taken from the last layer at [CLS] token, that is the first in the sequence order. This one feed, as usually, a linear classifier with dropout. The admitted sentiment label are $\{Positive, Negative, Conflict, Neutral\}$.
To select a label from the set, i have designed two different final linear layer for the network: one very standard with 4 neurons and a softmax function, and another one with two neuron and a sigmoid function. This last architecture work in this way: the first neuron represent positive label and the second the negative one , then if both neurons active at the same time the sentiment is classified as "conflict", otherwise, the sentiment is "neutral". Into final scores tables [2] and [4] these architecture are called *BIN* and *FOUR*.

4.3 Task C

Task C consist to extract, given a review, the categories that are related to the text. Each sample can belong, of course, to multiple category at the same time. The set of category is fixed, as opposed to task A: $\{service, food, miscellaneous, price, ambience\}$.
My approach to solve this task is very simple and consist to construct a multi-label classification model based on bert. Like the previous model, bert [CLS] output feed a linear classifier with dropout and a final sigmoid. In this case i decided to not try other experiment since the performances obtained from the very first time were excellent.

4.4 Task D

The approach for task D is similar to resolution strategy of task B. I exploit bert for a sentence pair classification fine-tuning, where, at this time, the first sentence is still the review, and the second one is the category. Performance are quite good and still comparable to task B.

4.5 End-To-End ABSA

After have found a good solution to all the different subtasks, i also tried a unified approach to merge

into one model, task A and B. This means to perform aspect based sentiment analysis in one shot, the model have to identify the target and assign the sentiment at the same time.
The model i designed to perform this, is similar to model A, but with a different tagging schema that can encapsulate the sentiment of the target. For this reason each token can be classified as { *Outside, Inside, B-Pos, B-Neg, B-Neut, B-Conf* }. In this case, instead of a linear layer, bert feeds an bi-lstm cell in a sequence to sequence configuration. Then the lstm hidden output will pass trough a linear layer and a final softmax. In this case, a recurrent architecture was chosen to give to the model more expressive power and help it to deal with more labels. The idea to implement a End-to-End solution came up in my mind after reading (Li et al., 2019).

5 Training Stage

As you have just read, i have tried different output encoding and model configuration. So, for each task, i select the best model picking the one with the highest macro-f1 score on dev-set. Instead, during the training of each model, i select and save from the epoch checkpoints, of course, the one with the lowest validation loss *(except for e2e)*.
In each model, i did not use bert only as feature extractor, so with freezed weights in training phase, but i fine tuned bert for the task.
Consequently, the hyper-parameters for the network were selected starting from the suggestion given from the bert paper authors. I used a batch size of 16 or 32, depending on the task, and a small learning rate of $2 * e^{-5}$ to be as smooth as possible. A good size of the batch has to allow most of the batches to contain at least a sample that belongs to the tiny class (e.g. *conflict*), to optimize macro-f1 score. To not waist all pre-trained information, and to prevent overfitting, i decided also to use a L2-Norm to regularize the model, of $1 * e^{-4}$. Considering the small learning rate, and the high L2 factor, i set the maximum number of epochs to 6 or 8, depending on the task. All the hyper-parameters can be read in table [6].

6 Results

Overall results are very confortable, considering the difficulty of ABSA. For task A, my model was able to achieve a f1 score of about 80 % with BIO tasg schema [1]. Simple IO tag schema perform a bit worse respect BIO. Anyway, maybe the best solution is to use CRF that could increase again these scores. For task B, my model reach a macro-f1 score of 58%, that considering the tiny class "conflict" is a very good result. In this case, output binary sentiment encoding was worse respect classic softmax configuration [2]. The same considerations are still applicable for model D. For task C my model was able to achieve 84% of f1 [3]. Putting togheter the models in a pipeline, results for task A+B are near **48%** macro-f1, and much more for the micro one. The end-to-end model is a few point below this score [5]. Finally, for C+D, i managed to get a score greater than **55%** [4]. These results were measured using *test.sh* script on both laptop and restaurant dev set, merged together in a unique file.

7 Conclusions

Aspect Based Sentiment Analysis is a very challenging task, because it is difficult and combines different subtask. Results of SOTA systems are nowadays anyway slightly far from being perfect, and also the solutions presented in this report were not really exciting in the overall performances. Anyway, from a my perspective, the model needs to be evaluated with micro-f1. Possible future improvements can exploit more the transformers architecture to gain some few extra point in performance.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. 2019. Exploiting BERT for end-to-end aspect-based sentiment analysis. *CoRR*, abs/1910.00883.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35. Association for Computational Linguistics.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 380–385. Association for Computational Linguistics.

Tag schema	Precision	Recall	F1-Score
IO	70.36	81.61	75.57
IOB	80.15	79.11	**79.63**

Table 1: **Task A** Results: Precision, Recall and F1-score for each tagging schema (and relative model).

Model	Precision		Recall		F1-Score	
	Macro	Micro	Macro	Micro	Macro	Micro
BIN (2)	49.62	70.24	48.56	69.98	48.61	70.11
FOUR (4)	59.52	75.88	57.71	75.60	**58.46**	75.74

Table 2: **Task B** Results: Precision, Recall and F1-score for each model. The two different architecture are explained in section 4.2, *BIN* and *FOUR* represents the number of out neurons. Macro and Micro scores are provided.

Precision		Recall		F1-Score	
Macro	Micro	Macro	Micro	Macro	Micro
86.80	87.00	80.31	82.81	**83.09**	84.85

Table 3: **Task C** Results: Precision, Recall and F1-score. Macro and Micro scores are provided.

Model	Precision		Recall		F1-Score	
	Macro	Micro	Macro	Micro	Macro	Micro
BIN (2)	51.43	64.18	50.15	61.09	49.96	62.60
FOUR (4)	58.42	67.04	54.19	63.68	**55.09**	65.38

Table 4: **Task C+D** Results: Precision, Recall and F1-score for each model, *BIN* and *FOUR* represents the number of out neurons of the models, like for task B. Macro and Micro scores are provided.

Model	Precision		Recall		F1-Score	
	Macro	Micro	Macro	Micro	Macro	Micro
Pipeline A+B	49.03	61.70	46.14	60.68	**47.23**	61.19
End-To-End	43.31	55.45	50.49	65.56	46.59	60.08

Table 5: **Task A+B** Results: Precision, Recall and F1-score for *end-to-end* architecture and the best combination of model for *task A and B* in sequential pipeline (see grey rows in previous tables). Macro and Micro scores are provided.

Task	Model	Max Ep.	Batch Size	L. Rate	L2-Norm	Dropout	Loss Weight
A	IO	6	32	0.0005	0.0001	0.2	No
A	IOB	6	32	0.0005	0.0001	0.2	No
B	BIN	6	16	0.0002	0.0001	0.2	No
B	FOUR	8	16	0.0002	0.0001	0.2	Yes
C	/	6	32	0.0005	0.0001	0.2	No
D	BIN	6	16	0.0002	0.0001	0.2	No
D	FOUR	8	16	0.0002	0.0001	0.2	Yes
E2E	/	6	16	0.0002	0.0001	0.2	Yes

Table 6: **Hyperparameter** selection for each task and model.

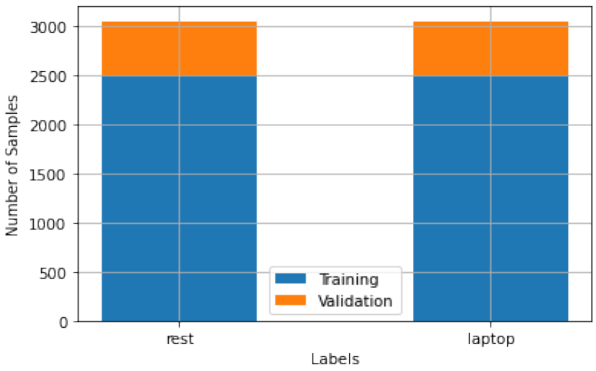

Figure 1: Sizes of train and dev set of restaurant and laptop datasets (comparison).

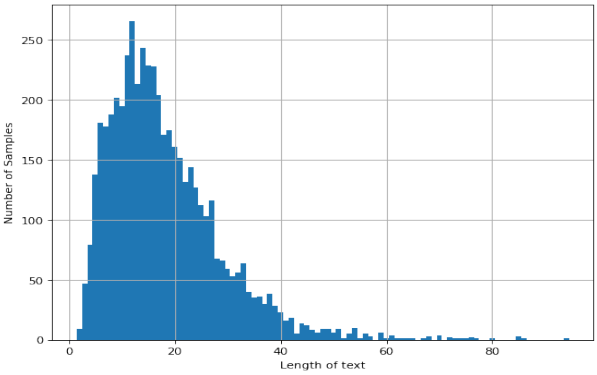

Figure 2: Distribution of the lengths of the sentences in the dataset. In the X-axis there is the number of token.

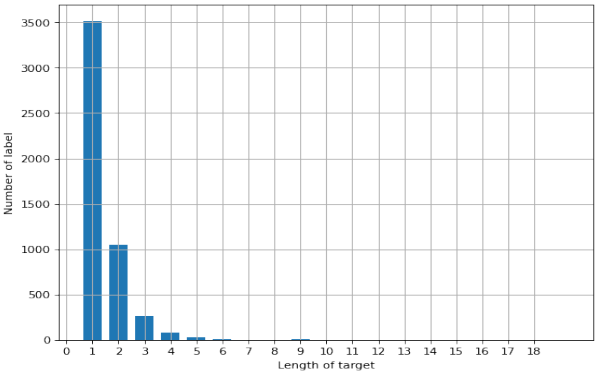

Figure 3: Distribution of the lengths of the aspect terms in the dataset. In the X-axis there is the number of token. A huge part of the target aspects have a length less or equal to 3 tokens.

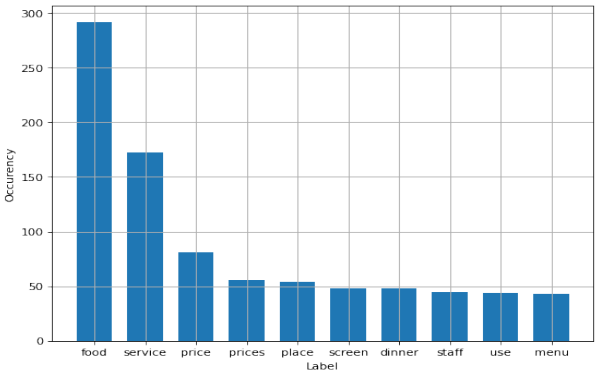

Figure 4: Histogram respect the most frequent aspect targets. They are mostly restaurant targets.

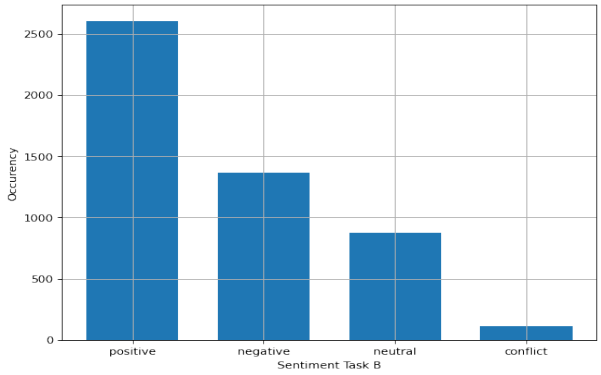

Figure 5: Task B - Sentiment classes sizes distribution.

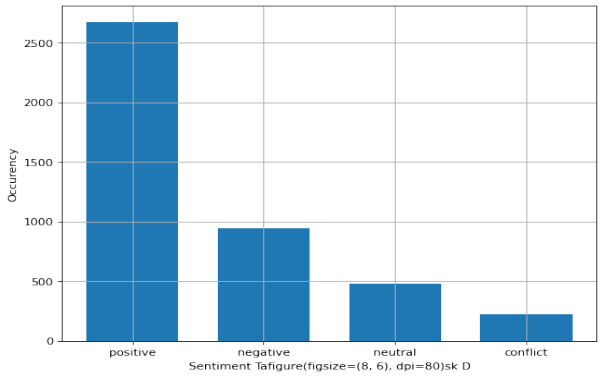

Figure 6: Task D - Sentiment classes sizes distribution.

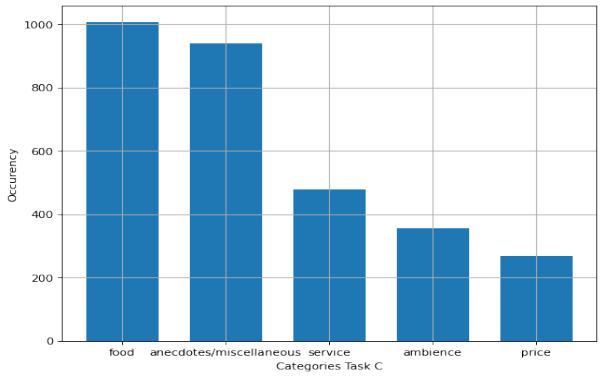

Figure 7: Task C - Category classes sizes distribution.