Reichman University

Efi Arazi School of Computer Science

M.Sc. Program - Research Track

Aspect Based Sentiment Analysis in Low-Resource Settings

by

Daniel Korat

M.Sc. dissertation, submitted in partial fulfillment of the requirements for the M.Sc. degree, research track, School of Computer Science Reichman University (The Interdisciplinary Center, Herzliya)

June 2022

This work was carried out under the supervision of Dr. Kfir Bar from the Efi Arazi School of Computer Science, Reichman University.

Abstract

A fundamental task of aspect-based sentiment analysis is aspect and opinion terms extraction. Supervised-learning approaches have shown good results for this task; however, they underperform in real-world settings where labeled data is lacking. Non pre-trained methods that incorporate external linguistic knowledge have proven effective in unsupervised domain adaptation settings; however, pre-trained transformer-based models like BERT and RoBERTa already exhibit substantial syntactic knowledge. We propose a method for incorporating external linguistic information into a self-attention mechanism coupled with BERT. This enables leveraging the intrinsic knowledge existing within BERT together with externally introduced syntactic information, to bridge the gap across domains. We demonstrate enhanced results with this method on three benchmark datasets. Another approach for low-resource scenarios is few-shot learning. Pattern-exploiting training has been shown to be effective in few-shot sequence classification. We design a method to use pattern-exploiting training for the token classification task of aspect term extraction. We demonstrate that this method significantly outperforms the standard supervised baseline in few-shot setups on three datasets.

Contents

Chapter 1

Introduction

Discovering and analysing opinions in user-generated content is crucial for widespread applications, including business review analysis, financial market prediction and political analysis (Yadav and Vishwakarma, 2020). Given the large scale of online content, it is intractable to manually process the opinion information. Therefore, an automatic computational solution for analyzing opinions in unstructured text is necessary. This has resulted in the emergence of the field of Sentiment Analysis (SA), or opinion mining.

Conventional SA operates at the sentence or document level, attributing one polarity value (positive, negative, neutral, etc.) to a whole sentence or document. It is assumed that a single overall sentiment is conveyed towards the single topic in the given text. However, opinionated text usually contains various emotional tendencies, expressed towards different aspects of entities. For example, in the sentence *"The battery life on this laptop is incredible but the performance is sluggish"*, the entity is the laptop, a positive sentiment is expressed towards the aspect *battery life* but a negative one is expressed towards the aspect *performance*. In consequence, the need for detecting more fine-grained sentiments at the aspect-level, termed Aspect Based Sentiment Analysis (ABSA), has received increasing attention in the past decade.

ABSA is the task of extracting, from a given corpus, aspect terms (opinion targets), and the sentiment expressed towards them. An aspect refers to a word or a phrase describing an aspect of an entity. A fundamental task of ABSA is aspect and opinion terms extraction (ATE and OTE, respectively). In the example sentence above, the aspect terms *battery life* and *performance* are associated with the opinion terms *incredible* and *sluggish*, respectively. Since ATE and OTE are crucial for ABSA, these tasks have also gained increasing attention, and featured on SemEval

shared tasks (Pontiki et al., 2014, 2015, 2016). Several ABSA datasets have been compiled, including SemEval Restaurant Review dataset and Laptop Review dataset (Pontiki et al., 2014), and Digital Device Review dataset (Hu and Liu, 2004a). These datasets have since become the main benchmark datasets for the ABSA task.

The fine-grained trait of ABSA makes it effective for generating a comprehensive detailed summary of opinion polarities expressed per aspect. Such information provides extremely valuable insight for businesses and consumers, since it enables measuring levels of satisfaction from different aspects of a product or service, based on a massive volume of consumer experiences. ABSA has been successfully applied to many domains and downstream tasks, including analysing reviews on laptops and restaurants (Karimi et al., 2021), hotels (Bajaj et al., 2021), education (Chauhan et al., 2019), product reviews on Twitter (Zainuddin et al., 2018), and more recently, social media trend analysis, like tracking COVID-19 discourse on Twitter (Jang et al., 2021). Furthermore, ABSA can be used to automate customer support tasks in response to user queries, and to improve recommendation systems (Da'u et al., 2020).

Most of the work related to aspect and opinion term extraction is formulated as a supervised token classification task. RNN-based models (Liu et al., 2015a) and Transformer-based (Vaswani et al., 2017) pre-trained language models (PLMs) showed promising results in single-domain setups when trained on thousands of labeled examples which come from the same domain as the test data. However, real-world scenarios are often required to handle multiple domains, while labeled training data per domain is scarce and costly. The supervised approaches typically do not scale across different domains, where only unlabeled data is available for the target domain, since aspect terms from two different domains are usually semantically different hence separated in the embedding space. For example, frequent aspect terms in the restaurant domain, like *salad* and *dessert*, have little or no semantic relatedness to frequent aspect terms in the laptop domain, like *screen size* and *battery life*.

This study offers two distinct strategies to address the problem. The first method (Chapter 2) tries to bridge the gap between domains by using auto-generated syntactic information in an unsupervised domain adaptation setting; that is, without requiring any labels from the target domain. The second method (Chapter 3) is a few-shot learning system, which leverages the

existing language modeling capability of PLMs and requires only a handful of in-domain labels.

1.1 Related Work

The most active line of ATE related work has been based on supervised learning, where token classification methods that utilize RNNs (Liu et al., 2015b), CRFs (Yin et al., 2016) and CNNs (Xu et al., 2018) have been proposed. Xu et al. (2019) proposed to post-train BERT (Devlin et al., 2019a), a pre-trained language model (PLM), on domain-specific data to obtain better word representations. Karimi et al. (2021) employed a CRF in conjunction with BERT using layer aggregation without fine-tuning. Although they achieve promising results, these methods require large amounts of labeled data, especially when training very sophisticated models.

There are several lines of work that address labeled data scarcity in ATE. One approach employs domain adaptation techniques, namely, utilizing existing labeled data from one domain to adapt a model to another domain. Ding et al. (2017) proposed using dependency-based aspect extraction rules as auxiliary supervision for an RNN model. However, this method depends on the quality of manually-crafted rules. Wang and Pan (2019) addressed this issue by designing a dependency prediction task that encodes dependency relations into the hidden representations of words, thus shifting the representations of different aspect terms having similar dependency relations, close to each other. Wang and Pan (2020) have further enhanced this model by integrating a conditional domain-adversarial network that encodes both word features and syntactic parent relation types. The above methods rely on the fact that syntactic information is important for identifying aspect and opinion terms (Hu and Liu, 2004b; Qiu et al., 2011).

This recent line of work demonstrates effective domain adaptation by incorporating syntactic knowledge into non pre-trained models during their training step. Subsequently, recent studies (Clark et al., 2019; Htut et al., 2019) show that pre-trained transformer-based models such as BERT and RoBERTa (Liu et al., 2019) already exhibit substantial linguistic knowledge. The model we present in Chapter 2 is designed to leverage both the syntactic information from pre-trained transformer models, and that from external sources, to further enhance domain adaptation in ATE.

Another approach for data scarcity in ATE, which is in its initial stages, is zero-shot ATE. Shu

et al. (2022) demonstrated that ATE can be successfully presented as a natural language inference task, achieving state-of-the-art results for zero-shot. However, the accuracy degradation compared to supervised methods is significant.

Recently, few-shot methods have demonstrated impressive results in **sequence classification** tasks. PET (Schick and Schütze, 2021a,b) is a recent approach that leverages patterns for few-shot learning, by reformulating natural language understanding tasks as cloze-style questions. ADAPET (Tam et al., 2021) modifies PET's objective to provide denser supervision during fine-tuning, alleviating the need for task-specific unlabeled data.

Subsequently, in Chapter 3 we propose a new framework to reformulate ATE, a **token classification** task, as a masked language modeling (MLM) task. As opposed to other ATE methods, this approach does not rely on out-of-domain labels, specialized neural architecture or handcrafted rules. Instead, we build upon the native token prediction capability of PLMs, combined with very few in-domain labels.

Chapter 2

Syntactically Aware Cross-Domain Aspect and Opinion Terms Extraction[1]

2.1 Introduction

This chapter examines whether the incorporation of external syntactic knowledge into pre-trained models contributes to bridging the gap across domains. For this purpose, we propose an approach for unsupervised domain-adaptation of aspect and opinion terms extraction based on incorporating linguistic knowledge into a pre-trained BERT model.

Specifically, inspired by Strubell et al. (2018), we incorporate externally-generated dependency relations into a self-attention mechanism that is coupled with the pre-trained BERT model (Stickland and Murray, 2019), where the external information is introduced during the fine-tuning and testing stages of the model.

2.2 Motivation and Background

Formally, the task of aspect and opinion terms extraction can be formulated as a sequence tagging task. The input is a sequence of tokens $X = \{x_1, x_2, .., x_n\}$ where the objective is to predict a corresponding sequence of labels $Y = \{y_1, y_2, ..., y_n\}$ with $y_i \in \{BA, IA, BO, IO, N\}$, where BA, BO, IA and IO represent a beginning of aspect/opinion and inside of aspect/opinion, respectively, and N represents all other tokens. The goal of unsupervised domain adaptation is to predict the token-level labels y_i^T of unlabeled target domain sentences $D^T = \{(x_i^T)\}$, given a set of labeled sentences from a source domain $D^S = \{(X_j^S, Y_j^S)\}$.

[1] Published as a short paper at COLING 2020 (Pereg et al., 2020).

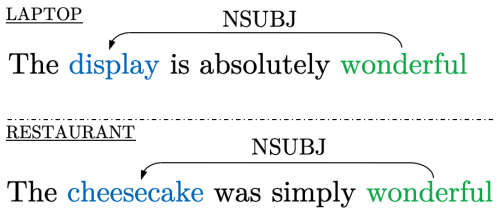

Figure 2.1: An example of opinionated sentences from two different domains with similar syntactic patterns. Opinion terms are colored green and aspect terms are colored blue.

It was observed that aspect and opinion terms maintain often-occurring syntactic patterns (Hu and Liu, 2004b; Qiu et al., 2011). Consider for example, a sentence from the laptop domain *"The display is absolutely wonderful"* and a sentence from the restaurant domain *"The cheesecake was simply wonderful"*. In the first sentence, an NSUBJ dependency relation exists between the opinion term (*'wonderful'*) and the aspect term (*'display'*). Assuming that the pattern aspect-NSUBJ-opinion is frequently observed in the laptop domain, then the term *cheesecake* can be extracted as an aspect term in the restaurant domain (Figure 2.1). This domain-independent trait of the syntactic structure can be leveraged for transferring knowledge from a labeled source domain to an unlabeled target domain. Recently, syntactic structure has been used for domain adaptation in non pre-trained models (see Chapter 1.1).

Analyses of pre-trained transformer-based models like BERT reveal substantial syntactic information captured within their attention mechanisms; however, those analyses also show that for many syntactic relations BERT only slightly improves over a simple baseline (Clark et al., 2019; Htut et al., 2019). Our goal is to design a neural network model that leverages both the information captured in the pre-trained model, and externally introduced syntactic information, to bridge the gap between the source and target domains.

2.3 The Proposed Model

The basis for our model is a pre-trained BERT-base model (Devlin et al., 2019b) with a fully connected sequence tagging classifier on top. Inspired by the work of Strubell et al. (2018), we incorporate dependency relations into a self-attention mechanism denoting a syntactically-aware attention head. Our approach differs from previous approaches which modify an existing self-attention head within a transformer-based model and train it from scratch. Our method modifies the BERT function by adding syntactically-aware self-attention heads in parallel to

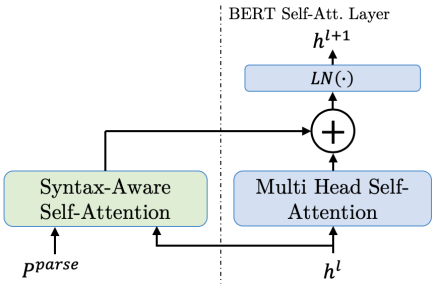

Figure 2.2: Coupling a syntactically-aware self-attention with a multi-head self-attention layer in a BERT model.

the BERT model (Stickland and Murray, 2019), and introduces the syntactic knowledge during the fine-tuning and testing stages. This leaves the original pre-trained model intact, enabling the model to utilize both the external linguistic information that is incorporated into the model and the intrinsic knowledge gained during the pre-training stage of the model. We refer to this model as syntactically-aware extended attention layer (SA-EXAL).

Multi-Head Self-Attention. The basis of our implementation is BERT's multi-head self-attention mechanism (Vaswani et al., 2017), which consists of I scaled dot-product attention heads. For each attention head i, the hidden token representations $h^l \in R^{d \times T}$, at the input of layer l, are projected to key, query and value representations K_i, Q_i and V_i of dimensions $T \times d_k$, where T is the number of tokens in the input sequence and $d_k = d/I$. Attention head i denotes attention weights that are a distinct distribution of every input token over all other tokens in the sequence:

$$A_i = softmax(\frac{Q_i K_i^T}{\sqrt{d_k}}) \tag{2.1}$$

The output of attention head i is denoted by $M_i = A_i V_i$, where M_i is a $T \times T$ matrix, in which each row t, represents a weighted sum of the value representations of all other tokens with respect to token t. Finally, the outputs of all I attention heads are concatenated and projected through a feed-forward(FF) network: $SA = FF(M_1, M_2, ..., M_I)$.

Syntactically-Aware Self-Attention. Inspired by the work of Strubell et al. (2018), we incorporate syntactic information into the self-attention head, forming a syntactically-aware self-attention, by encouraging it to attend to specific tokens corresponding to the syntactic structure of the sentence. As in the original attention-heads, we project h^l denoting K_{parse}, Q_{parse} and V_{parse} matrix representations of dimensions $T \times d_k$, but unlike the original heads, we also use

an external syntactic parser (Dozat and Manning, 2017) to generate P_{parse}, a $T \times T$ matrix in which each row t represents the probability of each token in the sentence to be the syntactic head of token t. We encourage this self-attention head to attend to the syntactic head of each token by performing an element-wise multiplication between P_{parse} and the dot product between the key and query matrices:

$$A_{parse} = softmax(\frac{(Q_{parse}K_{parse}^{T}) * P_{parse}}{\sqrt{d_k}}) \tag{2.2}$$

As in the original heads, The output of the syntactically-aware self-attention head is denoted by: $SA_{parse} = FF(A_{parse}V_{parse})$.

Adding Syntactically-Aware Self-Attention to BERT. Inspired by the work of Stickland and Murray (2019) we modify the BERT$(\cdot)$ function by adding a syntactically-aware self-attention head in parallel to each self-attention layer of the BERT model (see Figure 2.2) as follows:

$$h^{l+1} = LN(h^{l} + SA(h^{l}) + SA_{parse}(h^{l})) \tag{2.3}$$

where LN is BERT's layer normalization function and $h^{l} \in R^{d \times T}$ are the T hidden token representations at the input of layer l. Note that the contribution of the $SA_{parse}(h^{l})$ component to the representation of each token t in layer $l+1$, is mostly the representation of the syntactic head of token t. This shifts the representations of aspect terms from distinct domains, that syntactically relate to the same opinion term, closer to each other, thus contributing to bridging the gap between the domains.

2.4 Experiments

Data & Experimental Setup. Our experimental setup follows that of Wang and Pan (2020). We conduct experiments on benchmark datasets with customer reviews from three different domains: restaurant, laptop and digital devices. The restaurant domain combines reviews from SemEval 2014 (Pontiki et al., 2014) and SemEval 2015 (Pontiki et al., 2015). The laptop domain contains laptop reviews from SemEval 2014. Opinion term labels for these domains are ob-

Domain	# Sentences	Train	Dev.	Test
(R)estaurant	5,841	4,381	1,460	1,460
(L)aptop	3,845	2,884	961	961
(D)evice	3,836	2,877	959	959

Table 2.1: Sentence statistics for each domain.

	R → L		**R → D**		**L → R**		**L → D**		**D → R**		**D → L**	
Model	AS	OP	AS	OP	AS	OP	AS	OP	AS	OP	AS	OP
CrossCRF*	19.72 (1.82)	59.2 (1.34)	21.07 (0.44)	52.05 (1.67)	28.19 (0.58)	65.52 (0.89)	29.96 (1.69)	56.17 (1.49)	6.59 (0.49)	39.38 (3.06)	24.22 (2.54)	46.67 (2.43)
Hier-Joint*	33.66 (1.47)	- -	33.20 (0.52)	- -	48.10 (1.45)	- -	31.25 (0.49)	- -	47.97 (0.46)	- -	34.74 (2.27)	- -
RNCRF*	24.26 (3.97)	60.86 (3.35)	24.31 (2.57)	51.28 (1.78)	40.88 (2.09)	66.50 (1.48)	31.52 (1.40)	55.85 (1.09)	34.59 (1.34)	63.89 (1.59)	40.59 (0.80)	60.17 (1.20)
ARNN-GRU*	40.43 (0.96)	65.85 (1.50)	35.10 (0.62)	60.17 (0.75)	52.91 (1.82)	72.51 (1.03)	40.42 (0.70)	61.15 (0.60)	48.36 (1.14)	73.75 (1.76)	51.14 (1.68)	**71.18** (1.58)
TRNN-GRU*	40.15 (0.77)	65.63 (1.01)	37.33 (0.90)	60.32 (0.66)	53.78 (0.91)	73.40 (0.45)	41.19 (1.06)	60.20 (1.56)	51.17 (0.99)	**74.37** (1.03)	**51.66** (1.27)	68.79 (1.63)
EXAL	44.03 (2.11)	75.01 (1.13)	38.17 (0.79)	**63.59** (3.53)	48.23 (2.87)	79.57 (0.53)	41.60 (0.54)	**60.71** (5.49)	53.75 (1.24)	70.03 (2.46)	45.75 (1.54)	62.65 (2.51)
SA-EXAL	**47.59** (1.88)	**75.79** (1.02)	**40.50** (1.05)	63.33 (2.63)	**54.67** (2.02)	**80.05** (0.48)	**42.19** (0.54)	60.19 (3.79)	**54.54** (1.90)	71.57 (2.86)	47.72 (2.79)	63.98 (3.37)

Table 2.2: Comparison across different baselines in terms of average F1 scores (and standard variations in parentheses). *Results for non pre-trained baselines reported by (Wang and Pan, 2020). The best result for each dataset is highlighted in bold and the best result between EXAL and SA-EXAL is underlined.

tained from Wang et al. (2016). For the device domain, we use reviews from Hu and Liu (2004a) pertaining to five different digital products. Each token in each sentence is labeled as described in section 2.2. In order to make robust comparisons and to be comparable with previous work, for each domain we create three random splits of the data with a train/development/test ratio of 3:1:1 (see Table 2.1).

Since results may vary across random seeds (Dodge et al., 2020), we repeat each experiment using three different seeds and the final result is reported as the mean F1 score (and standard deviation) calculated over the three splits and the three seeds.

We adopt the HuggingFace (Wolf et al., 2019) implementation[2] of BERT-base (uncased) model as the basis for all experiments, and open-source our code.[3] We fine-tune the model with a learning rate of $5e^{-5}$, a batch size of 16 and a maximum sequence length of 64 tokens, for 10 epochs with an early stopping mechanism according to the development set. The dependency relations obtained by the Biaffine parser (Dozat and Manning, 2017) are generated in advance

[2] https://github.com/huggingface/transformers

[3] https://github.com/NervanaSystems/nlp-architect/tree/libert/nlp_architect/models/libert

and are introduced to the model during the fine-tuning as well as during the development/test stages. Following prior work, only exact matches between the predicted aspect and opinion terms and the gold labels are counted as correct.

Results. Table 2.2 shows a comparison of our proposed model (**SA-EXAL**) with notable baseline models, across different domain transfers. The baselines include:

- **CrossCRF** (Jakob and Gurevych, 2010): A linear-chain CRF with hand-engineered features (e.g. POS tags and dependencies).
- **Hier-Joint** (Ding et al., 2017): An RNN with auxiliary labels derived from manually designed rules that are based on frequently observed syntactic relations between aspect and opinion terms.
- **RNCRF** (Wang et al., 2016): A joint recursive neural network and CRF for in-domain aspect and opinion terms extraction.
- **ARNN-GRU** (Wang and Pan, 2020): A dependency-tree-based recursive neural network with GRU which uses an auto-encoder in the auxiliary task to reduce label noise.
- **TRNN-GRU** (Wang and Pan, 2020): An extension of ARNN-GRU which integrates a conditional domain-adversarial network that takes both word features and syntactic head relations as input.
- **EXAL**: A baseline model that shares the same size and structure as the proposed model SA-EXAL (Section 2.3) but does not incorporate syntactic information.

Our proposed model (SA-EXAL) shows an advantage over EXAL which demonstrate that although it was shown that the pre-trained BERT model captures significant linguistic knowledge, informing it with explicit external dependency relations is effective for transferring knowledge across domains. Specifically, SA-EXAL outperforms EXAL in 10 out of 12 cases (underlined in the table), including 6.44%, 3.56% and 2.33% improvements for L $\rightarrow$ R (AS), R $\rightarrow$ L (AS) and R $\rightarrow$ D (AS), respectively. We also note that SA-EXAL outperforms the non pre-trained model baselines in 8 out of 12 cases.

2.5 Conclusion

We propose a method for incorporating external linguistic information into a self-attention mechanism coupled with the BERT model. We demonstrate that this model leverages both

the intrinsic knowledge existing within the pre-trained model and the externally introduced syntactic information, to bridge the gap across domains.

Chapter 3

Few-Shot Aspect Term Extraction with Pattern-Exploiting Training[1]

3.1 Introduction

Our goal in this chapter is to mitigate the data scarcity challenge by reformulating the ATE task as a masked language modeling (MLM) task, which pre-trained language models are typically trained on, and excel in. We build upon a recent advancement in few-shot sequence classification titled pattern-exploiting training (PET) (Schick and Schütze, 2021a,b). PET represents input examples as cloze-style questions, which are then completed using language model predictions.

Our work is similar to PET in the sense that it employs pre-defined cloze patterns for MLM training, but it differs in the type of task; PET is designed for sequence classification, whereas our model is designed for token classification. This necessitates a new framework for mapping cloze phrase labels to token level annotations.

The contribution in this chapter is twofold. First, we propose a method for using pattern-exploiting training as cloze questions to address the task of aspect term extraction. Second, we show that this method significantly outperforms the standard PLM fine-tuning approach in few-shot scenarios.

[1]Submitted as a short paper to COLING 2022: Daniel Korat, Oren Pereg, Moshe Wasserblat, and Kfir Bar. 2022. Few-shot aspect term extraction with pattern-exploiting training. *Submitted to the 29th International Conference on Computational Linguistics*

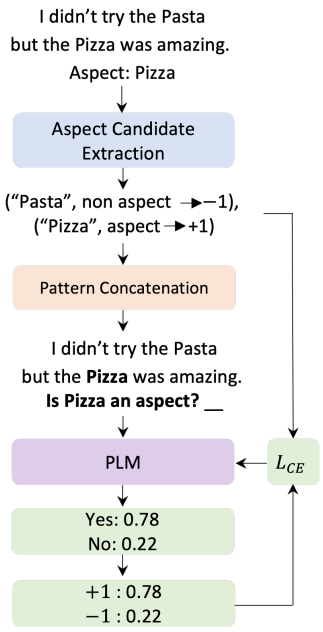

Figure 3.1: PET-ATE training: First, aspect candidates are extracted and are associated with labels according to the training set, then, yes/no cloze questions are concatenated and a PLM is trained to minimize the cross entropy loss (L_{CE}) from the correct answer.

3.2 PET-ATE

Figure 3.1 is an illustration of the PET-ATE method. First, aspect candidates are extracted from the input sentence. The candidates are associated with labels according to the training set. Next, yes/no cloze questions and their answers, aimed to qualify or disqualify the aspect candidates, are generated. Each cloze question is then concatenated to the input instance and finally, a PLM is fine-tuned to minimize the loss between the predicted answer and the correct answer.

3.2.1 Aspect Candidate Extraction

Cloze question patterns were shown to be effective in few-shot sequence classification (Schick and Schütze, 2021a; Tam et al., 2021; Zheng et al., 2022). However, token classification goal is more fine-grained. Our approach for exploiting cloze questions in ATE is based on introducing a pre-process mechanism that extracts aspect candidates. The aspect candidates are represented as text spans, that are of higher probability to function as aspects based on the context in which they appear. For this purpose we implement two aspect candidate extraction (ACE) methods. The first ACE method follows previous work that use noun and noun phrase detection for aspect candidate extraction (Hu and Liu, 2004b; Tulkens and van Cranenburgh, 2020). Along this line, we use a simple rule-based noun phrase extractor, based on patterns of part-of-speech

tags (Subhashini and Kumar, 2010; Chakraborty et al., 2016) for extracting aspect candidates. Formally, the output is denoted by $T^{nn} = \{x_i, asp_i^{nn}\}$ where x_i is a sentence and asp_i^{nn} is a noun phrase that constitutes an aspect candidate in the context of x_i. At training time, the final output of the ACE step is $T^C = T^{nn}$.

In order to complement the first ACE method with non-noun aspect candidates, we introduce a second ACE method, which operates at inference time. This method, based on a neural network, employs a PLM that is fine-tuned using the available labeled data for the ATE task, in a similar fashion to the baseline implementation (see Section 3.3). Then, this PLM is used to extract aspect candidates, generating $T^{neu} = \{x_i, asp_i^{neu}\}$ where x_i is a sentence and asp_i^{neu} is an aspect candidate. At inference time, the final output of the ACE step is produced by unifying both ACE methods, denoting $T^C = T^{nn} \cup T^{neu} = \{x_i, asp_i^c\}$.

3.2.2 Training Set Generation

Given a small set of labeled examples $T^G = \{x_i, asp_i^g\}$ where x_i is a sentence and asp_i^g is a gold aspect in the context of x_i, and a set of examples containing aspect candidates T^C, we generate a training set T by unifying the examples from T^G and T^C such that $T = T^G \cup T^C = \{x_i, asp_i, y_i\}$ where x_i is a sentence, asp_i is a span of text within x_i, and $y_i \in (-1, 1)$ indicates whether asp_i is an aspect $(y_i = +1)$ or a non aspect $(y_i = -1)$. We set y_i to be $+1$ for all the examples in T^G and set y_i to be -1 for all the examples that are not in the gold training set. Formally:

$$y_i = \begin{cases} +1 & asp_i \in T^G \\ -1 & asp_i \in (T^C - (T^C \cap T^G)) \end{cases}$$

3.2.3 Pattern Generation for Aspect Candidate Qualification

Given an input sentence x and a span of text asp in x, our goal is to predict whether asp is an aspect $(y = +1)$ or not $(y = -1)$ in the context of x. For this purpose we modify the PET (Schick and Schütze, 2021a,b) objective and define a function P that inputs a sentence x and a span of text asp and outputs a sentence $P(x, asp)$ that contains a yes/no question and exactly one mask token. We then define a verbalizer v that maps between the label y of

(x, asp) and a word in the PLM vocabulary, namely we set v = "Yes" if $(y = +1)$ which indicates that asp is indeed an aspect, and set v = "No" if $(y = -1)$ which indicates non aspect. For example, given the following input pair: (x, asp) = (The dessert was amazing, *dessert*), The task is redefined as concatenating to x a question asking whether the most likely choice in the masked position regarding asp (dessert) is "Yes" or "No", denoting: $P(x, asp)$ = The dessert was amazing. Does the review focus on *dessert*? __

3.2.4 Training and Inference

For each triplet (x, asp, y) in training set T, we generate a pair (p, v) where $p = P(x, asp)$, and v is the label verbalizer. We adapt the score for label y given input x generated by a PLM for a masked token defined by Schick and Schütze (2021a,b) to incorporate asp, denoting:

$$S_p(y|(x, asp)) = PLM(v(y)|P(x, asp)) \tag{3.1}$$

We then use the cross entropy between the softmaxed probability distribution of $S_p(y|(x, asp))$ and the true distribution of the training example summed over all the training examples $\{x_i, asp_i, y_i\}$ in T to fine-tune the PLM. Similarly, at inference time, Equation 3.1 is used to classify candidates as aspects/non-aspects by calculating their yes/no verbalizer scores. Following studies that show benefits of continued pretraining (CPT) of PLMs in general (Howard and Ruder, 2018; Gururangan et al., 2020) and in few-shot setups (Schick and Schütze, 2021a), we use unlabeled examples from the domain of the labeled data to train the PLM with an MLM objective, prior to fine-tuning.

3.3 Experiments

Datasets. Following previous ATE work, we conduct experiments on datasets of customer reviews from three different domains: Restaurant ($\mathbb{R}$), Laptop ($\mathbb{L}$) and Device ($\mathbb{D}$). $\mathbb{R}$ includes restaurant reviews from SemEval 2014 (Pontiki et al., 2014) and SemEval 2015 (Pontiki et al., 2015). $\mathbb{L}$ includes laptop reviews from SemEval2014, and $\mathbb{D}$ is provided by Hu and Liu (2004a) and contains reviews of five different digital products.

Experimental Setup. Systematically evaluating few-shot performance can be challenging, as fine-tuning using small datasets may incur instability (Dodge et al., 2020; Zhang et al., 2021), and results may change dramatically given different random data selections. Thus, we adopt a rigorous framework for training and evaluating few-shot methods, proposed by Zheng et al. (2022). Each dataset is first split into D_{test} $(1/3)$ and D_{train} $(2/3)$. To tune the model for N labeled training examples, we select N examples from D_{train}, denoted D_N. Then, we apply a multi-split strategy, wherein D_N is randomly divided into equally sized D^k_{train} and D^k_{dev}. This division process is repeated $K = 5$ times. Given a hyper-parameter space H, for each $h \in H$ and $k \in 1...5$, we train PET-ATE on D^k_{train} using h and evaluate it on D^k_{dev}. Let h^* be the hyper-parameter set that achieves the best mean F1-score across all k splits. To test the model performance, we train it using h^* on D_N, and evaluate on D_{test}. Finally, we report the mean F1-score for this test over 3 random seeds. The selected h^* per dataset appear in Table A.4 in Appendix A.2. CPT always uses the full D_{train} (unlabeled), containing **2,565**, **3,896** and **2,557** examples for $\mathbb{L}$, $\mathbb{R}$ and $\mathbb{D}$, respectively. We adopt the HuggingFace implementation[2] of RoBERTa-base (Liu et al., 2019), with a modified training objective (Section 3.2.4).

Baseline. Our standard supervised training baseline is based on the common approach of formulating the ATE task as a token classification task (Poria et al., 2016; Xu et al., 2018) by fine-tuning the same RoBERTa-base model with a token classification layer using the few-shot labeled data. For fair comparison, we tune the baseline hyper-parameters in the same manner performed for PET-ATE.

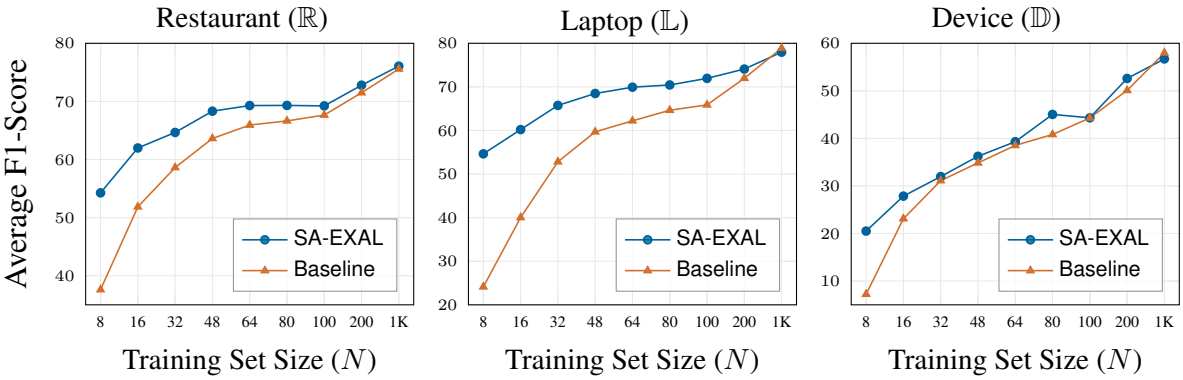

Figure 3.2: F1 as a function of the training set size N of SA-EXAL and the baseline model on $\mathbb{R}$, $\mathbb{L}$ and $\mathbb{D}$

[2] https://github.com/huggingface/transformers

3.4 Results

Figure 3.2 shows a comparison between PET-ATE and the baseline for different values of N; PET-ATE significantly outperforms the baseline in small N values. For example, for $N = 8$, PET-ATE outperforms the baseline by **16.6**, **30.5**, and **13.3** average F1 on $\mathbb{R}$, $\mathbb{L}$ and $\mathbb{D}$, respectively. As N increases, the performance gains decrease, however PET-ATE outperforms the baseline for $N \leq 1000$ in $\mathbb{R}$ and for $N \leq 200$ in $\mathbb{L}$. Detailed results are shown in Tables A.1-A.3 in Appendix A.1. These results highlight the model's ability to better leverage the inherent knowledge of the PLM, by using cloze-style patterns during training.

N	P_0	P_1	P_2	P_3
8	**54.6** ± 2.7	53.3 ± 6.4	53.6 ± 2.6	51.3 ± 6.2
16	60.2 ± 1.1	62.2 ± 0.8	**62.8** ± 1.4	61.4 ± 0.5
32	**65.8** ± 1.1	64.4 ± 0.9	64.4 ± 0.9	62.7 ± 1.3
64	69.9 ± 1.2	68.7 ± 0.5	69.4 ± 1.1	**70.0** ± 0.5
100	**72.0** ± 0.3	70.2 ± 1.3	70.8 ± 1.2	70.7 ± 1.9
200	**74.1** ± 0.7	73.2 ± 0.9	**74.1** ± 0.5	**74.1** ± 1.1
1000	78.0 ± 0.5	**78.2** ± 0.5	78.0 ± 0.4	**78.2** ± 0.8

Table 3.1: Average F1 and standard deviation for PET-ATE on Laptops dataset using different cloze-patterns, for various training set sizes N

N	**Method**	$\mathbb{L}$	$\mathbb{R}$
16	Baseline	40.0 ± 7.3	51.9 ± 0.9
	PET-ATE w/o n-ACE*	56.1 ± 3.7	55.1 ± 5.2
	PET-ATE w/o CPT**	51.7 ± 4.4	56.5 ± 4.7
	PET-ATE	**60.2** ± 1.1	**62.0** ± 1.2
64	Baseline	62.2 ± 1.6	65.9 ± 1.1
	PET-ATE w/o n-ACE*	64.2 ± 0.9	64.2 ± 1.3
	PET-ATE w/o CPT**	65.5 ± 3.6	65.6 ± 1.1
	PET-ATE	**69.9** ± 1.1	**69.3** ± 1.6
128	Baseline	68.0 ± 2.2	69.6 ± 0.4
	PET-ATE w/o n-ACE*	65.3 ± 0.1	66.7 ± 0.3
	PET-ATE w/o CPT**	71.5 ± 1.1	68.4 ± 1.7
	PET-ATE	**72.4** ± 0.4	**69.7** ± 1.9

Table 3.2: PET-ATE ablation test showing average F1 and standard deviation for the Laptop and Restaurant datasets across three training set sizes N. *PET-ATE excluding the neural ACE step. **PET-ATE excluding the continued pre-training step (Section 3.2.4).

We evaluated PET-ATE using four different cloze patterns (see details in Appendix A.3). The results in Table 3.1 show that pattern selection has a very small effect on the performance and there was no single prominent pattern. In fact, using the null-pattern P_3, composed only of the candidate aspect followed by a masked Yes/No token, yielded F1 comparable to other patterns.

This demonstrates the robustness of PET-ATE for pattern selection.

To study the effect of the neural ACE and CPT on the model's performance, we conducted ablation experiments across three values of N. The results, in Table 3.2, show that both steps hold a significant contribution, especially for $N \leq 64$. CPT is most valuable at $N = 16$, and its contribution decreases as N increases. The gains from neural ACE are significant across all tested values for N.

3.5 Conclusion

We propose a method for using pattern-exploiting training in the form of cloze questions for few-shot aspect term extraction. We demonstrate that this method leverages the inherent masked token prediction trait of PLMs and outperforms the standard supervised training baseline in few-shot setups. This enables easy adaptation to new domains where labeled data is scarce.

Chapter 4

Conclusion

The main obstacle in aspect based sentiment analysis is the high cost of token-level labeling. This study proposes two paths to tackle this data deficiency problem. Both paths combine novel techniques with intrinsic knowledge present in pre-trained language models (implicit syntactic knowledge or native word completion capability).

The first path relies on existing out-of-domain labels. We point out that syntactic relations are important in aspect and opinion term extraction, and that pre-trained language models exhibit this syntactic knowledge. Our main contribution is a method for incorporating external syntactic information into a pre-trained language model, to bridge the gap across domains.

The second path uses a few-shot strategy; leveraging few labels from the target domain. We design a method to reformulate aspect term extraction as a cloze question task, which is where pre-trained language model excel. Second, we demonstrate that this method significantly outperforms the standard language model fine-tuning approach in few-shot scenarios. This enables easy adaptation to new domains where labeled data is scarce.

Syntactic features are still used in state-of-the-art methods, strengthening our initial hypothesis. For example, Chen and Qian (2021), which cite our method in Chapter 2, encode syntactic information (part-of-speech tags and dependency relation types) into a single trainable vector. However, we believe that future work should focus on assessing the confidence of predictions, partly due to syntactic errors. Moreover, both strategies in this work can be improved by reducing their model size, and extending them to other token-level tasks such as aspect polarity classification and named entity recognition.

Bibliography

Vaibhav Bajaj, Kartikey Pant, Ishan Upadhyay, Srinath Nair, and Radhika Mamidi. 2021. TEASER: Towards efficient aspect-based SEntiment analysis and recognition. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 102–110, Held Online. INCOMA Ltd.

Neelotpal Chakraborty, Samir Malakar, Ram Sarkar, and Mita Nasipuri. 2016. A rule based approach for noun phrase extraction from english text document. In *Seventh International Conference on CNC*, pages 13–26.

Ganpat Singh Chauhan, Preksha Agrawal, and Yogesh Kumar Meena. 2019. Aspect-based sentiment analysis of students' feedback to improve teaching–learning process. In *Information and communication technology for intelligent systems*, pages 259–266. Springer.

Zhuang Chen and Tieyun Qian. 2021. Bridge-based active domain adaptation for aspect term extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 317–327, Online. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Aminu Da'u, Naomie Salim, Idris Rabiu, and Akram Osman. 2020. Recommendation system exploiting aspect-based opinion mining with deep learning method. *Information Sciences*, 512:1279–1292.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ying Ding, Jianfei Yu, and Jing Jiang. 2017. Recurrent neural networks with auxiliary labels for crossdomain opinion target extraction. In *Association for the Advancement of Artificial Intelligence*, pages 3436—3442.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Phu Mon Htut, Jason Phang, Shikha Bordia, and Samuel R. Bowman. 2019. Do attention heads in bert track syntactic dependencies? In *arXiv*.

Minqing Hu and Bing Liu. 2004a. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Minqing Hu and Bing Liu. 2004b. Mining opinion features in customer reviews. In *American Association for Artificial Intelligence*.

Niklan Jakob and Iryna Gurevych. 2010. Extracting opinion targets in a single- and cross-domain setting with conditional random fields. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1035–1045.

Hyeju Jang, Emily Rempel, David Roth, Giuseppe Carenini, and Naveed Zafar Janjua. 2021. Tracking covid-19 discourse on twitter in north america: Infodemiology study using topic modeling and aspect-based sentiment analysis. *Journal of medical Internet research*, 23(2):e25431.

Akbar Karimi, Leonardo Rossi, and Andrea Prati. 2021. Improving BERT performance for aspect-based sentiment analysis. In *Proceedings of The Fourth International Conference on Natural Language and Speech Processing (ICNLSP 2021)*, pages 39–46, Trento, Italy. Association for Computational Linguistics.

Pengfei Liu, Shafiq Joty, and Helen Meng. 2015a. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1433–1443, Lisbon, Portugal. Association for Computational Linguistics.

Pengfei Liu, Shafiq Joty, and Helen Meng. 2015b. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1433–1443, Lisbon, Portugal. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. Cite arxiv:1907.11692.

Oren Pereg, Daniel Korat, and Moshe Wasserblat. 2020. Syntactically aware cross-domain aspect and opinion terms extraction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1772–1777, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado. Association for Computational Linguistics.

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, Dublin, Ireland. Association for Computational Linguistics.

Soujanya Poria, Erik Cambria, and Alexander Gelbukh. 2016. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, 108:42–49. New Avenues in Knowledge Bases for Natural Language Processing.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2011. Opinion word expansion and target extraction through double propagation. *Comput. Linguist.*, 37(1):9–27.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021b. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6943–6951, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lei Shu, Jiahua Chen, Bing Liu, and Hu Xu. 2022. Zero-shot aspect-based sentiment analysis. *arXiv preprint arXiv:2202.01924*.

Asa Cooper Stickland and Iain Murray. 2019. BERT and PALs: Projected attention layers for efficient adaptation in multi-task learning. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5986–5995, Long Beach, California, USA. PMLR.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5027–5038, Brussels, Belgium. Association for Computational Linguistics.

R Subhashini and V Jawahar Senthil Kumar. 2010. Shallow nlp techniques for noun phrase extraction. In *Trendz in Information Sciences & Computing (TISC2010)*, pages 73–77. IEEE.

Derek Tam, Rakesh R. Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. 2021. Improving and simplifying pattern exploiting training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4980–4991, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Stéphan Tulkens and Andreas van Cranenburgh. 2020. Embarrassingly simple unsupervised aspect extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3182–3187, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Wenya Wang and Sinno Jialin Pan. 2019. Syntactically meaningful and transferable recursive neural networks for aspect and opinion extraction. *Computational Linguistics*, 45(4):705–736.

Wenya Wang and Sinno Jialin Pan. 2020. Syntactically meaningful and transferable recursive neural networks for aspect and opinion extraction. *Computational Linguistics*, 45(4):705–736.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in NaturalLanguage Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 616–626.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. 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 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2018. Double embeddings and CNN-based sequence labeling for aspect extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 592–598, Melbourne, Australia. Association for Computational Linguistics.

Ashima Yadav and Dinesh Kumar Vishwakarma. 2020. Sentiment analysis using deep learning architectures: a review. *Artificial Intelligence Review*, 53(6):4335–4385.

Yichun Yin, Furu Wei, Li Dong, Kaimeng Xu, Ming Zhang, and Ming Zhou. 2016. Unsupervised word and dependency path embeddings for aspect term extraction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 2979–2985. AAAI Press.

Nurulhuda Zainuddin, Ali Selamat, and Roliana Ibrahim. 2018. Hybrid sentiment classification on twitter aspect-based sentiment analysis. *Applied Intelligence*, 48(5):1218–1232.

Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q Weinberger, and Yoav Artzi. 2021. Revisiting few-sample {bert} fine-tuning. In *International Conference on Learning Representations*.

Yanan Zheng, Jing Zhou, Yujie Qian, Ming Ding, Chonghua Liao, Li Jian, Ruslan Salakhutdinov, Jie Tang, Sebastian Ruder, and Zhilin Yang. 2022. FewNLU: Benchmarking state-of-the-art methods for few-shot natural language understanding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 501–516, Dublin, Ireland. Association for Computational Linguistics.

Appendix A

Chapter 3 Appendix

A.1 Full results

We provide average precision, recall and F1 scores (and their standard deviations), per dataset, in Tables A.1-A.3. The overall relative advantage in recall can be attributed in part to the ACE step, due to its two different strategies for discovering aspect candidates (POS features and token classification) which may complement each other.

N	**Method**	**P**	**R**	**F1**
8	Baseline	53.1 ± 8.8	31.1 ± 9.3	37.6 ± 7.1
8	PET-ATE	53.6 ± 5.9	56.7 ± 8.1	54.3 ± 0.3
16	Baseline	47.6 ± 1.6	57.2 ± 4.0	51.9 ± 0.9
16	PET-ATE	54.7 ± 3.9	72.2 ± 5.0	62.0 ± 1.2
32	Baseline	57.4 ± 5.3	61.5 ± 9.3	58.6 ± 3.8
32	PET-ATE	61.1 ± 5.7	70.2 ± 8.7	64.7 ± 2.3
48	Baseline	59.5 ± 4.2	68.6 ± 1.0	63.6 ± 2.1
48	PET-ATE	64.2 ± 2.0	73.0 ± 2.8	68.3 ± 1.9
64	Baseline	62.0 ± 1.4	70.4 ± 2.0	65.9 ± 1.1
64	PET-ATE	67.1 ± 1.5	71.7 ± 1.4	69.3 ± 0.2
80	Baseline	64.7 ± 2.7	69.0 ± 3.1	66.7 ± 0.5
80	PET-ATE	68.1 ± 0.7	70.7 ± 3.1	69.3 ± 1.5
100	Baseline	63.6 ± 2.2	72.4 ± 1.3	67.7 ± 0.7
100	PET-ATE	67.0 ± 1.9	71.7 ± 1.1	69.2 ± 1.4
200	Baseline	68.8 ± 0.7	74.4 ± 0.4	71.5 ± 0.5
200	PET-ATE	70.7 ± 1.5	75.1 ± 1.6	72.8 ± 0.8
1000	Baseline	74.7 ± 1.0	76.5 ± 0.8	75.6 ± 0.2
1000	PET-ATE	74.4 ± 0.5	77.8 ± 0.3	76.1 ± 0.3

Table A.1: Average Precision (P), Recall (R), F1 and standard deviation for the baseline and PET-ATE on Restaurant dataset for various training set sizes N.

N	Method	P	R	F1
8	Baseline	34.7 ± 6.4	19.5 ± 5.2	24.1 ± 3.5
8	PET-ATE	55.9 ± 9.7	55.7 ± 8.0	54.6 ± 2.7
16	Baseline	38.3 ± 8.4	42.6 ± 7.5	40.0 ± 7.3
16	PET-ATE	59.4 ± 5.5	62.5 ± 7.8	60.2 ± 1.1
32	Baseline	51.1 ± 0.6	54.8 ± 3.0	52.8 ± 1.7
32	PET-ATE	63.4 ± 2.3	68.4 ± 1.8	65.8 ± 1.1
48	Baseline	54.2 ± 0.9	64.0 ± 1.9	58.7 ± 1.0
48	PET-ATE	65.8 ± 1.5	71.6 ± 3.5	68.5 ± 1.5
64	Baseline	60.4 ± 2.5	64.2 ± 1.5	62.2 ± 1.6
64	PET-ATE	67.7 ± 3.7	72.7 ± 3.6	69.9 ± 1.2
80	Baseline	64.0 ± 0.6	65.4 ± 2.9	64.6 ± 1.3
80	PET-ATE	68.7 ± 2.1	72.4 ± 3.1	70.4 ± 0.9
100	Baseline	64.8 ± 1.8	67.3 ± 3.7	65.9 ± 1.0
100	PET-ATE	70.0 ± 1.5	74.1 ± 1.6	72.0 ± 0.3
200	Baseline	71.2 ± 0.6	72.8 ± 2.2	72.0 ± 1.4
200	PET-ATE	72.8 ± 1.2	75.4 ± 0.2	74.1 ± 0.7
1000	Baseline	76.6 ± 0.6	81.3 ± 0.3	78.9 ± 0.2
1000	PET-ATE	76.0 ± 0.6	80.1 ± 1.6	78.0 ± 0.5

Table A.2: Average Precision (P), Recall (R), F1 and standard deviation for the baseline and PET-ATE on Laptop dataset for various training set sizes N.

N	Method	P	R	F1
8	Baseline	26.8 ± 24.8	4.2 ± 3.8	7.2 ± 6.7
8	PET-ATE	31.7 ± 6.8	16.2 ± 4.5	20.5 ± 3.4
16	Baseline	38.6 ± 11.1	17.3 ± 6.3	23.1 ± 6.1
16	PET-ATE	38.0 ± 11.3	23.1 ± 2.7	27.9 ± 4.2
32	Baseline	35.1 ± 8.5	28.8 ± 10.4	31.1 ± 8.8
32	PET-ATE	41.4 ± 4.5	26.8 ± 8.3	32.0 ± 7.2
48	Baseline	41.1 ± 8.7	30.7 ± 3.9	34.9 ± 5.0
48	PET-ATE	41.6 ± 2.5	32.4 ± 6.7	36.2 ± 5.1
64	Baseline	45.9 ± 3.2	33.3 ± 0.5	38.5 ± 0.9
64	PET-ATE	45.3 ± 0.1	35.0 ± 4.8	39.3 ± 3.1
80	Baseline	47.0 ± 3.0	36.8 ± 6.1	40.8 ± 3.0
80	PET-ATE	50.7 ± 1.4	40.7 ± 4.5	45.0 ± 3.3
100	Baseline	47.9 ± 1.4	41.5 ± 4.2	44.3 ± 1.7
100	PET-ATE	48.7 ± 2.8	40.8 ± 1.7	44.3 ± 0.9
200	Baseline	55.5 ± 5.1	45.7 ± 3.2	50.1 ± 3.8
200	PET-ATE	52.2 ± 4.5	53.3 ± 1.6	52.6 ± 1.6
1000	Baseline	56.0 ± 1.2	60.2 ± 1.6	58.0 ± 1.3
1000	PET-ATE	53.5 ± 1.7	60.4 ± 1.8	56.7 ± 0.6

Table A.3: Average Precision (P), Recall (R), F1 and standard deviation for the baseline and PET-ATE on Device dataset for various training set sizes N.

A.2 Hyper-parameters

Hyper-parameter tuning was performed as described in Section 3.3. For the baseline, we try values in the range $[1e{-}5, 3e{-}5]$ for the `learning_rate`, $[400, 2000]$ for the number of training steps (`max_steps`) and $[8, 16]$ for the batch size. For PET-ATE, the search values are:

`max_steps` $\in \{700, 1000\}$,

`learning_rate` $\in \{2e{-}5, 3e{-}5\}$,

Parameter	Baseline			PET-ATE		
	$\mathbb{L}$	$\mathbb{R}$	$\mathbb{D}$	$\mathbb{L}$	$\mathbb{R}$	$\mathbb{D}$
adam_epsilon	1e-8	1e-8	1e-8	1e-8	1e-8	1e-8
max_seq_length	128	128	128	128	128	128
mlm_probability	–	–	–	0.15	0.15	0.15
learning_rate	3e-5	3e-5	3e-5	2e-5	3e-5	2e-5
per_device_train_batch_size	16	16	16	8	8	8
max_steps	600	800	1000	700	1000	1000
neural_ace_max_steps	–	–	–	500	1000	500
neural_ace_learning_rate	–	–	–	2e-5	2e-5	2e-5
CPT_max_steps	–	–	–	1000	2000	2000
CPT_learning_rate	–	–	–	2e-5	3e-5	2e-5

Table A.4: Hyperparameters for baseline and PET-ATE per dataset

`neu_ace_max_steps` $\in \{500, 1000\}$,

`neu_ace_learning_rate` $\in \{2\text{e}{-5}, 3\text{e}{-5}\}$,

`CPT_max_steps` $\in \{1000, 2000\}$,

`CPT_learning_rate`$\in \{1\text{e}{-5}, 2\text{e}{-5}, 3\text{e}{-5}\}$.

We provide the selected parameters for each setup, in Table A.4. Note that the batch size is always 8 where not specified.

A.3 Patterns

We tested cloze patterns with varying content words, word order and length. Table 3.1 shows results for P_0 – the pattern used for the primary tests – as well as 3 additional patterns P_1, P_2, P_3. P_3 is a null-pattern, that is, it does not contain any tokens other than the candidate aspect followed by a masked Yes/No token. Given an input sentence x_i and a span of text asp_i in x_i, these are the patterns:

$P_0(x, asp^c) = x$. So, does the review in the previous sentence focus on asp^c? __

$P_1(x, asp^c) = x$. Is asp^c an aspect? __

$P_2(x, asp^c) = x$. So, is the review about asp^c? __

$P_3(x, asp^c) = x\ asp^c$ __

תקציר

המשימה הבסיסית ביותר בניתוח סנטימנט מבוסס היבט היא חילוץ מונחי היבט ומונחי דעה. גישות הנוקטות בלמידה מונחית הדגימו תוצאות טובות במשימה זו ; עם זאת, הן משיגות תוצאות ירודות בתרחישים מהעולם האמיתי בהם יש מחסור במידע מתויג. שיטות המשלבות ידע לשוני חיצוני הוכחו כיעילות בתרחישים של אדפטציית דומיינים בלתי מונחית. אולם, שיטות אלו אינן מבוססות על מודלים מאומנים מראש. מודלי שפה מאומנים מראש כדוגמת BERT ו-RoBERTa כבר מגלמים בתוכם מידע תחבירי ממשי. אנו מציעים שיטה לשילוב מידע לשוני חיצוני במנגנון attention המוצמד ל-BERT. כך, ניתן למנף את הידע המוטמע ב-BERT בצירוף עם מידע תחבירי מבחוץ, כדי לגשר על הפער בין דומיינים. אנו מדגימים תוצאות משופרות עם שיטה זו על גבי שלושה מאגרי מידע. גישה נוספת בתרחישים דלי משאבים היא למידה ממיעוט דוגמאות. למידה מבוססת תבניות הוכחה כיעילה בסיווג רצפי מילים עם מיעוט דוגמאות. אנו מציגים שיטה לשימוש בלמידה מבוססת תבניות לטובת משימת סיווג מילים בודדות – חילוץ מונחי היבט. בבדיקה על גבי 3 מאגרי מידע, שיטה זו עולה בביצועיה על שיטת הבסיס הסטנדרטית בתרחישים מעוטי דוגמאות.

עבודה זו בוצעה בהדרכתו של ד״ר כפיר בר מבי״ס אפי ארזי למדעי המחשב, אוניברסיטת רייכמן.

אוניברסיטת רייכמן
בית-ספר אפי ארזי למדעי המחשב
התכנית לתואר שני (M.Sc.) - מסלול מחקרי

ניתוח סנטימנט מבוסס היבט בתרחיש דל משאבים

מאת
דניאל קורת

עבודת תזה המוגשת כחלק מהדרישות לשם קבלת תואר מוסמך M.Sc.
במסלול המחקרי בבית ספר אפי ארזי למדעי המחשב, אוניברסיטת רייכמן

יוני 2022