Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 4933–4941
Marseille, 11–16 May 2020

Adapt or Get Left Behind: Domain Adaptation through BERT Language Model Finetuning for Aspect-Target Sentiment Classification

Alexander Rietzler, Sebastian Stabinger, Paul Opitz, Stefan Engl
DeepOpinion
Bozner Platz 1, Innsbruck, Austria
{alexander.rietzler, sebastian.stabinger, paul.opitz, stefan.engl}@deepopinion.ai

Abstract

Aspect-Target Sentiment Classification (ATSC) is a subtask of Aspect-Based Sentiment Analysis (ABSA), which has many applications e.g. in e-commerce, where data and insights from reviews can be leveraged to create value for businesses and customers. Recently, deep transfer-learning methods have been applied successfully to a myriad of Natural Language Processing (NLP) tasks, including ATSC. Building on top of the prominent BERT language model, we approach ATSC using a two-step procedure: self-supervised domain-specific BERT language model finetuning, followed by supervised task-specific finetuning. Our findings on how to best exploit domain-specific language model finetuning enable us to produce new state-of-the-art performance on the SemEval 2014 Task 4 restaurants dataset. In addition, to explore the real-world robustness of our models, we perform cross-domain evaluation. We show that a cross-domain adapted BERT language model performs significantly better than strong baseline models like vanilla BERT-base and XLNet-base. Finally, we conduct a case study to interpret model prediction errors.

Keywords: Aspect-Based Sentiment Analysis, Targeted Sentiment Classification, Cross-Domain Adaptation, BERT, XLNet

1. Introduction

Sentiment Analysis (SA) is an active field of research in Natural Language Processing and deals with opinions in text. A typical application of classical SA in an industrial setting would be to classify a document like a product review into *positive, negative* or *neutral* sentiment polarity.

In contrast to SA, the more fine-grained task of Aspect-Based Sentiment Analysis (ABSA) (Hu and Liu, 2004; Pontiki et al., 2015) aims to find both the aspect of an entity like a restaurant, and the sentiment associated with this aspect. It is important to note that ABSA comes in two variants. We will use the sentence *"I love their dumplings"* to explain these variants in detail.

Both variants are implemented as a two-step procedure. The first variant is comprised of Aspect-Category Detection (ACD) followed by Aspect-Category Sentiment Classification (ACSC). ACD is a multilabel classification task, where a sentence can be associated with a set of predefined aspect categories like *"food"* and *"service"* in the restaurants domain. In the second step, ACSC, the sentiment polarity associated to the aspect-category is classified. For our example-sentence the correct result is the tuple *("food", "positive")*.

The second variant consists of Aspect-Target Extraction (ATE) followed by Aspect-Target Sentiment Classification (ATSC). ATE is a sequence labeling task, where terms like *"dumplings"* are detected. In the second step, ATSC, the sentiment polarity associated with the aspect-target is determined. In our example the correct result is *("dumplings", "positive")*.

In this paper, we focus on ATSC. In recent years, specialized neural architectures (Tang et al., 2016a; Tang et al., 2016b) have been developed that substantially improved modeling of this target-context relationship. More recently, the Natural Language Processing community experienced a substantial shift towards using pre-trained language models (Peters et al., 2018; Radford and Salimans, 2018; Howard and Ruder, 2018; Devlin et al., 2019) as a base for many down-stream tasks, which also includes ABSA (Song et al., 2019; Xu et al., 2019; Sun et al., 2019). We still see huge potential that comes with this trend, which is why we approach the ATSC task using the BERT architecture.

As shown by Xu et al. (2019), for the ATSC task the performance of models that were pre-trained on general text corpora is improved substantially by finetuning the language model on domain-specific corpora — in their case review corpora — that have not been used for pre-training BERT, or other language models.

We extend the work by Xu et al. by further investigating the behavior of finetuning the BERT language model in relation to ATSC performance. In particular, our contributions are:

1. Analysis of the influence of the amount of training-steps used for BERT language model finetuning on the Aspect-Target Sentiment Classification performance.
2. Findings on how exploiting BERT language model finetuning enables us to achieve new state-of-the-art performance on the SemEval 2014 restaurants dataset.
3. Analysis of cross-domain adaptation between the laptops and restaurant domains. Adaptation is tested by self-supervised finetuning of the BERT language model on the target-domain and then supervised training on the ATSC task in the source-domain. In addition, the performance of training on the combination of both datasets is measured.

2. Related Works

We separate our discussion of related work into two areas: first, neural methods applied to ATSC that have improved

performance solely by model architecture improvements. Secondly, methods that additionally aim to transfer knowledge from semantically related tasks or domains.

2.1. Architecture Improvements for Aspect-Target Sentiment Classification

The datasets typically used for Aspect-Target Sentiment Classification are the SemEval 2014 Task 4 datasets (Pontiki et al., 2015) for the restaurants and laptops domain. Both datasets have only a small number of training examples. One common approach to compensate for insufficient training examples is to invent neural architectures that better model ATSC. For example, in the past a big leap in classification performance was achieved with the use of the Memory Network architecture (Tang et al., 2016b), which uses memory to remember context words and explicitly models attention over both the target word and context. It was found that making full use of context words improves their model compared to previous models (Tang et al., 2016a) that make use of left- and right-sided context independently.

Song et al. (2019) proposed Attention Encoder Networks (AEN), a modification to the transformer architecture. The authors split the Multi-Head Attention (MHA) layers into Intra-MHA and Inter-MHA layers in order to model target words and context differently, which results in a more lightweight model compared to the transformer architecture.

Another recent performance leap was achieved by Zhaoa et al. (2019), who model dependencies between sentiment words explicitly in sentences with more than one aspect-target by using a graph convolutional neural network. They show that their architecture performs particularly well if multiple aspects are present in a sentence.

2.2. Knowledge Transfer for Aspect-Target Sentiment Classification Analysis

One approach to compensate for insufficient training examples is to transfer knowledge across domains or across similar tasks.

Li et al. (2019) proposed Multi-Granularity Alignment Networks (MGAN). They use this architecture to transfer knowledge from both an aspect-category classification task and also across different domains. They built a large scale aspect-category dataset specifically for this.

He et al. (2018) transfer knowledge from a document-level sentiment classification task trained on the Amazon review dataset (He and McAuley, 2016). They successfully apply pre-training by reusing the weights of a Long Short Term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997) that has been trained on the document-level sentiment task. In addition, they apply multi-task learning where aspect and document-level tasks are learned simultaneously by minimizing a joint loss function.

Similarly, Xu et al. (2019) introduce a multi-task loss function to simultaneously optimize on BERT model's (Devlin et al., 2019) pre-training objectives as well as a question answering task.

In contrast to the methods described above that aim to transfer knowledge from a different source task - like question answering or document-level sentiment classification - this paper aims at transferring knowledge across different domains by self-supervised finetuning of the BERT language model.

3. Methodology

We approach the Aspect-Target Sentiment Classification task using a two-step procedure. We use the pre-trained BERT architecture as a basis. In the first step we finetune the pre-trained weights of the language model further in a self-supervised way on a domain-specific corpus. In the second step we train the finetuned language model in a supervised way on the ATSC end-task.

In the following subsections, we discuss the BERT architecture, how we finetune the language model, and how we transform the ATSC task into a BERT sequence-pair classification task (Sun et al., 2019). Subsequently, we discuss all the different end-task training and domain-specific finetuning combinations we employ to evaluate our model's generalization performance not only in-domain but also cross-domain.

Finally, we describe how we apply input reduction, an interpretation method for neural NLP models, to the ATSC task.

3.1. BERT

The BERT model builds on many previous innovations: contextualized word representations (Peters et al., 2018), the transformer architecture (Vaswani et al., 2017), and pre-training on a language modeling task with subsequent end-to-end finetuning on a downstream task (Radford and Salimans, 2018; Howard and Ruder, 2018). Due to being deeply bidirectional, the BERT architecture creates powerful sequence representations that perform extremely well on many downstream tasks (Devlin et al., 2019).

The main innovation of BERT is that instead of using the objective of next-word prediction, a different objective is used to train the language model. This objective consists of two parts.

The first part is the masked language model objective, where the model learns to predict randomly masked tokens from their context.

The second part is the next-sequence prediction objective, where the model needs to predict if a sequence B would naturally follow the previous sequence A. This objective enables the model to capture long-term dependencies better. Both objectives are discussed in more detail in the next section.

As a base for our experiments we use the $\text{BERT}_{\text{BASE}}$ model, which has been pre-trained by the Google research team. It has the following parameters: 12 layers, 768 hidden dimensions per token and 12 attention heads. It has 110 million parameters in total.

For finetuning the BERT language model on a specific domain we use the weights of $\text{BERT}_{\text{BASE}}$ as a starting point.

3.2. BERT Language Model Finetuning

As the first step of our procedure we perform language model finetuning of the BERT model using domain-specific corpora. Algorithmically, this is equivalent to pre-training.

The domain-specific language model finetuning as an intermediate step to ATSC has been described by Xu et al. (2019). As an extension to their paper we investigate the limits of language model finetuning in terms of how end-task performance is dependent on the amount of training steps.
The training input representation for language model finetuning consists of two sequences s_A and s_B in the format of "[CLS] s_A [SEP] s_B [SEP]", where [CLS] is a dummy token used for downstream classification and [SEP] are separator tokens.

Masked Language Model Objective
The sequences A and B have tokens randomly masked out in order for the model to learn to predict them. The following example shows how domain-specific finetuning could alleviate the bias from pre-training on a Wikipedia corpus: *"The touchscreen is an* [MASK] *device"*. In the fact-based context of Wikipedia the [MASK] could be *"input"* and in the review domain a typical guess could be the general opinion word *"amazing"*.

Next-Sentence Prediction
In order to train BERT to capture long-term dependencies better, the model is trained to predict whether sequence B follows sequence A. If this is the case, sequence A and sequence B are jointly sampled from the same document in the order they appear naturally. Otherwise the sequences are sampled randomly from the training corpus.

3.3. Aspect-Target Sentiment Classification

The ATSC task aims at classifying sentiment polarity into the three classes *positive, negative, neutral* with respect to an aspect-target. The input to the classifier is a tokenized sentence $s = s_{1:n}$ and a target $t = s_{j:j+m}$ contained in the sentence, where $j < j + m \leq n$. Similar to previous work by Sun et al. (2019), we transform the input into a format compatible with BERT sequence-pair classification tasks: "[CLS] s [SEP] t [SEP]".
In the BERT architecture the position of the token embeddings is structurally maintained after each Multi-Head Attention layer. Therefore, we refer to the last hidden representation of the [CLS] token as $h_{[CLS]} \in \mathbf{R}^{768\times1}$. The number of sentiment polarity classes is three. A distribution $p \in [0, 1]^3$ over these classes is predicted using a fully-connected layer with 3 output neurons on top of $h_{[CLS]}$, followed by a softmax activation function

$$p = \text{softmax}(W \cdot h_{[CLS]} + b),$$

where $b \in \mathbf{R}^3$ and $W \in \mathbf{R}^{3\times768}$. Cross-entropy is used as the training loss. The way we use BERT for classifying the sentiment polarities is equivalent to how BERT is used for sequence-pair classification tasks in the original paper (Devlin et al., 2019).

3.4. Domain Adaptation through Language Model Finetuning

In academia, it is common that the performance of a machine learning model is evaluated *in-domain*. This means that the model is evaluated on a test set that comes from the same distribution as the training set. In real-world applications this setting is not always valid, as the trained model is used to predict previously unseen data.
In order to evaluate the performance of a machine learning model more robustly, its generalization error can be evaluated across different domains, i.e. *cross-domain*. To optimize cross-domain performance, the model itself can be adapted towards a target domain. This procedure is known as Domain Adaptation, which is a special case of Transductive Transfer Learning in the taxonomy of Ruder (2019). Here, it is typically assumed that supervised data for a specific task is only available for a *source domain* S, whereas only unsupervised data is available in the *target domain* T. The goal is to optimize performance of the task in the target domain while transferring task-specific knowledge from the source domain.
If we map this framework to our challenge, we define Aspect-Target Sentiment Classification as the transfer-task and BERT language model finetuning is used for domain adaptation. In terms of which domain is finetuned on, the full transfer-procedure can be expressed in the following way:

$$D_{LM} \rightarrow D_{Train} \rightarrow D_{Test}.$$

Here, D_{LM} stands for the domain on which the language model is finetuned and can take on the values of *Restaurants*, *Laptops* or *(Restaurants ∪ Laptops)*. The domain for training D_{Train} can take on the same values; for the joint case the training datasets for laptops and restaurants are simply combined. The domain for testing D_{Test} can only take the value *Restaurants* or *Laptops*.
Combining finetuning and training steps gives us nine different evaluation scenarios, which we group into the following four categories:

In-Domain Training
ATSC is trained on a domain-specific dataset and evaluated on the test set from the same domain. This can be expressed as
$D_{LM} \rightarrow T \rightarrow T$, where T is our target domain and can be either *Laptops* or *Restaurants*. It is expected that the performance of the model is highest if $D_{LM} = T$.

Cross-Domain Training
ATSC is trained on a domain-specific dataset and evaluated on the test set from the other domain. This can be expressed as
$D_{LM} \rightarrow S \rightarrow T$, where $S \neq T$ are source and target domain and can be either *Laptops* or *Restaurants*.

Cross-Domain Adaptation
As a special case of cross-domain training we expect performance to be optimal if $D_{LM} = T$. This is the variant of *Domain Adaptation* and is written as
$T \rightarrow S \rightarrow T$.

Joint-Domain Training
ATSC is trained on both domain-specific datasets jointly and evaluated on both test sets independently. This can be expressed as

$D_{LM} \rightarrow (S \cup T) \rightarrow T$, where $S \neq T$ are source- and target domain and can either be *Laptops* or *Restaurants*.

3.5. Input Reduction for Model Interpretation

Input reduction is an interpretation method for neural models introduced by Feng et al. (2018), which tries to find a subset of the most important words of a document that contribute most to a prediction.

We use this interpretation method to illustrate the predictions of our models on the test set in order to find potential causes for classification errors, and also to find qualitative differences between our models and baseline models.

The input reduction method resembles a process that iteratively removes unimportant words from the input while the model's prediction is maintained. The idea is that the remaining set of words one iteration before the prediction flips are the most important ones. As pointed out by Feng et al. (2018) for this method to work, a machine learning model needs to compute meaningful confidence values for unseen input. For our task, we find empirically that the predicted probabilities computed for our test set examples work well enough as a confidence approximation, which means that most of the reduced input for the examples discussed in subsection 4.5. allows for a meaningful interpretation.

Let $\mathbf{x} = [x_1, x_2, \ldots x_n]$ be the input sentence represented as a list of tokens and $p(y|\mathbf{x})$ the predicted probability of label y, and $y = \text{argmax}_{\hat{y}}\ p(\hat{y}|\mathbf{x})$ the originally predicted label. The importance of a word is defined as

$$g(x_i) = p(y|\mathbf{x}) - p(y|\mathbf{x}_{-i}).$$

Put differently, the importance of a word is the prediction probability towards the original label of a sentence containing the word minus the prediction probability of the sentence without the same word.

We apply this formula to iteratively remove the word with the lowest importance until the prediction changes to another label. Due to the nature of the ATSC task, we make an exception for words that are part of the aspect-target phrase, which we do not remove during an iteration. This allows us to maintain the context with respect to the aspect-target.

4. Experiments

In our experiments we aim to answer the following research questions (RQs):

RQ1: How does the number of training iterations in the BERT language model finetuning stage influence the ATSC end-task performance? At what point does performance start to improve, when does it converge?

RQ2: If trained in-domain, what ATSC end-task performance can be reached through fully exploited finetuning of the BERT language model?

RQ3: If trained cross-domain in the special case of domain adaptation, what ATSC end-task performance can be reached if BERT language model finetuning is fully exploited?

4.1. Datasets for Classification and Language Model Finetuning

We conduct experiments using the two SemEval 2014 Task 4 Subtask 2 datasets[1] (Pontiki et al., 2015) for the laptops and the restaurants domain. The two datasets contain sentences with one or multiple marked aspect-targets that each have a 3-level sentiment polarity (*positive, neutral or negative*) associated. In the original dataset the *conflict* class is also present. Here, the *conflict* labels are dropped for reasons of comparability with Xu et al. (2019). Detailed statistics for both datasets are shown in Table 1.

For BERT language model finetuning we prepare three corpora for the two domains of laptops and restaurants. For the restaurants domain we use Yelp Dataset Challenge reviews[2] and for the laptops domain we use Amazon Laptop reviews (He and McAuley, 2016). For the laptop domain we filtered out reviews that appear in the SemEval 2014 laptops dataset to avoid training bias for the test data. To be compatible with the next-sentence prediction task used during fine tuning, we removed reviews containing fewer than two sentences from the corpora.

For the laptop corpus, $1,007,209$ sentences are left after pre-processing. For the restaurants domain, where more reviews are available, we sampled $10,000,000$ sentences to have a sufficient amount of data for fully exploited language model finetuning. In order to compensate for the smaller amount of finetuning data in the laptops domain, we finetune for more epochs, 30 epochs in the case of the laptops domain compared to 3 epochs for the restaurants domain, so that the BERT model trains on about 30 million sentences in both cases. This means that a single sentence can appear multiple times with a different language model masking.

We also create a mixed corpus to jointly finetune on both domains. Here, we sample 1 million restaurant reviews and combine them with the laptop reviews. This results in about 2 million reviews that are finetuned for 15 epochs. The exact statistics for the three finetuning corpora are shown in the top of Table 1.

We release code to reproduce the generation of our finetuning corpora[3].

4.2. Hyperparameters

We use $\text{BERT}_{\text{BASE}}$[4] (uncased) as the base for all of our experiments, with the exception of $\text{XLNet}_{\text{BASE}}$ (cased), which is used as one of the baseline models.

For the BERT language model finetuning we use 32 bit floating point computations using the Adam optimizer (Kingma and Ba, 2014). The batchsize is set to 32 while the learning rate is set to $3 \cdot 10^{-5}$. The maximum input sequence length is set to 256 tokens, which amounts to about 4 sentences per sequence on average. As shown in

[1] http://alt.qcri.org/semeval2014/task4

[2] https://www.yelp.com/dataset/challenge

[3] https://github.com/deepopinion/domain-adapted-atsc

[4] We make use of both BERT-base-uncased and XLNet-base-cased models as part of the pytorch-transformers library: https://github.com/huggingface/pytorch-transformers

Corpus	Sentences	Finetuning Epochs
Laptops	1,007,209	30
Restaurants	10,000,000	3
Lapt.+Rest.	2,007,213	15

Dataset	Positive		Negative		Neutral	
	Train	Test	Train	Test	Train	Test
Laptops	987	341	866	128	460	169
Restaurants	2,164	728	805	196	633	196

Table 1: Top: Detailed statistics of the corpora for BERT language model finetuning. Bottom: Number of labels for each category of the SemEval 2014 Task 4 Subtask 2 laptop and restaurant datasets for Aspect-Target Sentiment Classification.

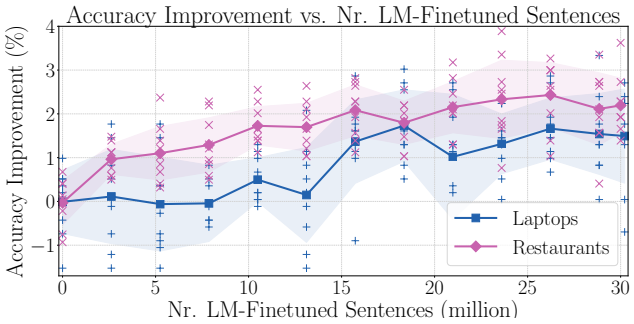

Figure 1: Absolute accuracy improvement of Aspect-Target Sentiment Classification as a function of the number of sentences the BERT language model has been finetuned on. Markers (■, ♦) connected through the lines are the averages (μ) over 9 runs, a single run is marked as either a cross ($\times$) for restaurants or a plus ($+$) for laptops. The standard deviation (σ) curves are also drawn ($\mu \pm \sigma$). The model is trained on the SemEval 2014 Task 4 datasets and evaluated in-domain. The language models are finetuned on the target-domain corpora. Best viewed in color.

Table 1, we finetune the language models on each domain so that the model trains a total of about 30 million sentences ($\approx$ 7.5 million sequences).
For training the BERT and XLNet models on the down-stream task of ATSC we use mixed 16 bit and 32 bit floating point computations, the Adam optimizer, and a learning rate of $3 \cdot 10^{-5}$ and a batchsize of 32. We train the model for a total of 7 epochs. The validation accuracy converges after about 3 epochs of training on all datasets, but training loss still improves after that.
It is important to note that all our results reported are the average of 9 runs with different random initializations. This is needed to measure significance of improvements, as the standard deviation in accuray amounts to roughly 1% for all experiments (see Figure 1).

4.3. Compared Methods

We compare in-domain results to current state-of-the-art methods, which we will now describe briefly.
SDGCN-BERT (Zhaoa et al., 2019) explicitly models sentiment dependencies for sentences with multiple aspects with a graph convolutional network. This method is currently state-of-the-art on the SemEval 2014 laptops dataset.
AEN-BERT (Song et al., 2019) is an attentional encoder network. When used on top of BERT embeddings this method performs especially well on the laptops dataset.
BERT-SPC (Song et al., 2019) is BERT used in sentence-pair classification mode. This is exactly the same method as our BERT-base baseline and therefore, we can cross-check the authors' results.
BERT-PT (Xu et al., 2019) uses multi-task fine-tuning prior to downstream classification, where the BERT language model is finetuned jointly with a question answering task. It has state-of-the-art performance on the restaurants dataset prior to this paper.
To our knowledge, cross- and joint-domain training on the SemEval 2014 Task 4 datasets has not been analyzed so far. Thus, we compare our method to two very strong baseline models: BERT-base and XLNet-base.
BERT-base (Devlin et al., 2019) is using the pre-trained $\text{BERT}_{\text{BASE}}$ embeddings directly on the down-stream task without any domain specific language model finetuning.
XLNet-base (Yang et al., 2019) is a method also based on general language model pre-training similar to BERT. Instead of randomly masking tokens for pre-training like BERT, a more general permutation objective is used, where all possible variants of masking are fully exploited.
Our models are BERT models whose language model has been finetuned on different domain corpora.
BERT-ADA Lapt is the BERT language model finetuned on the laptop domain corpus.
BERT-ADA Rest is the BERT language model finetuned on the restaurant domain corpus.
BERT-ADA Joint is the BERT language model finetuned on the corpus containing an equal amount of laptops and restaurants reviews.

4.4. Results Analysis

The results of our experiments are shown in Figure 1 and Table 2 respectively.
To answer RQ1, which is concerned with details of domain-specific language model finetuning, we can see in Figure 1 that first of all, language model finetuning has a significant effect on ATSC end-task performance. Secondly, we see that in the restaurants domain the performance starts to increase immediately, whereas in the laptops domain it takes about 10 million finetuned sentences before a significant increase can be measured. After around 17 million sentences no significant improvement can be measured. In addition, we find that the different runs have a high variance, which necessitates averaging over 9 runs to measure differences in model performance reliably.
To answer RQ2, which is concerned with in-domain ATSC performance, we see in Table 2 that for the in-domain training case, our models BERT-ADA Lapt and BERT-ADA Rest achieve performance close to state-of-the-art on the laptops dataset and new state-of-the-art on the restaurants dataset with accuracies of 79.19% and 87.14%, respectively. On the restaurants dataset, this corresponds to an absolute improvement of 2.2% compared to the previous state-of-the-art method BERT-PT. Language model fine-

Test Dataset	Laptops						Restaurants					
Train Dataset / **Train Type**	**Laptops** **In →**		**Restaurants** **Cross ↔**		**Lapt. + Rest.** **Joint ∪**		**Restaurants** **In →**		**Laptops** **Cross ↔**		**Lapt. + Rest.** **Joint ∪**	
Other Methods	**Acc**	**MF1**	**Acc**	**MF1**	**Acc**	**MF1**	**Acc**	**MF1**	**Acc**	**MF1**	**Acc**	**MF1**
SDGCN-BERT	**81.35**	**78.34**	-	-	-	-	83.57	76.47	-	-	-	-
AEN-BERT	79.93	76.31	-	-	-	-	83.12	73.76	-	-	-	-
BERT-SPC	78.99	75.03	-	-	-	-	84.46	76.98	-	-	-	-
BERT-PT	78.07	75.08	-	-	-	-	84.95	76.96	-	-	-	-
Baselines												
XLNet-base	79.89	77.78	77.78	72.24	**80.88**	76.92	85.84	78.35	82.41	72.98	86.15	78.93
BERT-base	77.69	72.60	75.86	70.78	78.81	74.47	84.92	76.93	80.07	69.93	85.03	77.35
Ours												
BERT-ADA Lapt	79.19	74.18	**77.92**	**72.99**	80.23	75.77	85.51	78.09	80.68	72.93	86.22	79.79
BERT-ADA Rest	78.60	74.09	76.16	70.46	79.14	74.93	**87.14**	**80.05**	**83.68**	72.91	**87.89**	81.05
BERT-ADA Joint	78.96	74.18	75.91	69.84	79.94	**78.74**	86.35	78.89	82.23	**73.03**	87.69	**81.20**

Table 2: Summary of results for Aspect-Target Sentiment Classification for in-domain, cross-domain, and joint-domain training on SemEval 2014 Task 4 Subtask 2 datasets. The cells with gray background correspond to the cross-domain adaptation case, where the language model is finetuned on the target domain. As evaluation metrics accuracy (Acc) and Macro-F1 (MF1) are used.

tuning produces a larger improvement on the restaurants dataset. We think that one reason for that might be that the restaurants domain is underrepresented in the pre-training corpora of $\text{BERT}_{\text{BASE}}$. Generally, we find that language model finetuning helps even if the finetuning domain does not match the evaluation domain. We think the reason for this might be that the BERT-base model is pre-trained more on knowledge-based corpora like Wikipedia than on text containing opinions. We show some evidence for this hypothesis in subsection 4.5.. In addition, we find that the XLNet-base baseline performs generally stronger than BERT-base, but only outperforms the BERT-ADA models on the laptops dataset with an accuracy of 79.89% .

To answer RQ3, which is concerned with domain adaptation, we can see from the grayed out cells in Table 2, which correspond to the cross-domain adaption case where the BERT language model is trained on the target domain, that domain adaptation works well with 2.2% absolute accuracy improvement on the laptops test set and even 3.6% accuracy improvement on the restaurants test set compared to BERT-base.

In general, the ATSC task generalizes well cross-domain, with about a 2-3$\%$ drop in accuracy compared to in-domain training. We think the reason for this might be that syntactical relationships between the aspect-target and the phrase expressing sentiment polarity, as well as knowing the sentiment-polarity itself, are sufficient to solve the ATSC task in most cases.

For the joint-training case, we find that combining both training datasets improves performance on both test sets. This result is intuitive, as more training data generally leads to better performance if the domains do not confuse each other. Interestingly, for the joint-training case the BERT-ADA Joint model performs especially well when measured by the Macro-F1 metric. A reason for this might be that the SemEval 2014 datasets are imbalanced due to dominance of positive labels. It seems like through finetuning the language model on both domains the model learns to classify the neutral class much better, especially in the laptops domain.

4.5. Case Study

The goal of the case study is to find answers to the following questions:

- What are potential causes for the improved performance of the finetuned language models BERT-ADA Lapt and BERT-ADA Rest over BERT-base based on analyzing cases that have differing sentiment predictions?
- Based on interpreting samples with incorrect predictions, what are potential reasons for these erroneous classifications?
- What error types prevent us from performing at human expert level on ATSC?

To answer these questions we performed input reduction, which allows for a better interpretation of sample predictions from the SemEval 2014 Restaurant and Laptops test set, see Table 3. The input reduction technique tries to isolate a set of words from the sentence that contribute most to the prediction. The theoretical details of input reduction are discussed in subsection 3.5..

Samples predicted correctly solely by the target-domain adapted model

In the following, we will discuss a selection of examples that are classified correctly by the best performing in-domain BERT-ADA, but incorrectly by BERT-base. The error types for BERT-base are mentioned for all the examples next to their reference label.

Ref.	Restaurant Samples	Aspect-Target	Base	Lapt	Rest	Gold
RC1	the icing_{L} MADE this cake, it was fluffy_{R}, not_{B} ultra sweet_{B}, creamy and light.	cake	–	+	+	+
RC2	The staff $\text{should}_{L,R}$ be a bit more_{L} friendly_{B}.	staff	+	–	–	–
RC3	$\text{15\%}_{B}$ gratuity $\text{automatically}_{R,L}$ added to the bill_{R}.	gratuity	+	+	–	–
RE1	My friend had_{L} a burger and I had these $\text{wonderful}_{B,R}$ blueberry pancakes.	burger	o	+	+	o
RE2	The sauce is $\text{excellent}_{B,L,R}$ (very fresh) with dabs of real mozzarella.	dabs of real mozzarella	+	+	+	o
	Laptop Samples					
LC1	The Mac mini is about 8x smaller than my old computer which is a huge bonus and runs_{B} $\text{very quiet}_{B,L}$, actually_{B} the fans $\text{aren't audible}_{R}$ unlike my old pc	fans	–	+	–	+
LE1	the latest version does $\text{not}_{B,R,L}$ have a disc drive.	disc drive	–	–	–	o
LE2	Which it did $\text{not}_{B,R}$ have, only_{L} 3 USB 2 ports.	USB 2 ports	–	–	–	o

Table 3: Shown are text samples from SemEval 2014 Restaurants and Laptops test-set that are predicted correctly for the language model adapted to the target domain but predicted falsely with the bert-base model (RC1-5, LC1-LC2). In addition, samples which are predicted falsely by the target-domain adapted model are shown (RE1-2, LE1-2). The abbreviations stand for: B – BERT-base, L – BERT-ADA Lapt(op), R – BERT-ADA Rest(aurant) – all the language models used for prediction. The used down-stream-classifiers are trained in-domain. The reduced input (set of words that influence prediction strongest) is formatted with underline and the subscript denotes the corresponding model (B, L, R) used for computing the reduced input. If viewed in color, the corresponding predicted sentiment polarity of the reduced input corresponds to: green – positive, red – negative, gray – neutral, alternating green and red – both negative and positive for different models. Best viewed in color.

RC1 – general review-domain context needed:
"not ... sweet" – this negated phrase is detected with dominant negative sentiment by BERT-base and seems to have an overwhelming influence on the prediction of the whole sentence, while ignoring all the other positive sentiment carrying words like fluffy, creamy and light. In contrast, for BERT-ADA Rest the expression *"fluffy"* carries the dominant positive sentiment resulting in a correct prediction of this sentence. Although BERT-ADA Lapt also predicts the positive sentiment correctly the dominant sentiment carrying phrase *"icing"* raises questions on how trustworthy and robust this prediction might be under minor changes of the sentence.
RC2 – general review-domain context needed:
We believe that *"should be"* is an expression often found in text containing opinions. This could be one of the main reasons that both BERT-ADA Lapt and Rest, which both have been finetuned on review-specific text, predict this example correctly. BERT-base is strongly influenced by *"friendly"* and does not detect the sentiment-negating function of *"should be"*.
RC3 – restaurant-domain context needed:
The reduced input *"gratuity"* is detected as positive for the BERT-ADA Laptop and BERT-base model, which makes sense if this word is presented in isolation. In contrast, the BERT-ADA Rest model reveals reduced input words *"automatically"* and *"bill"* with negative sentiment. This seems to indicate awareness on the relevant context, namely that its probably not positive for the consumer – usually the writer of a review – if gratuity is added automatically to his/her bill.
LC1 – laptop-domain context needed:
"very quiet" is classified as negative by BERT-base whereas the same expression is classified positive by the BERT-ADA Lapt model. This example is very interesting as it seems to indicate that the knowledge "quiet fans are something positive for its user", is solely extracted by finetuning the BERT on the laptop domain. Same reasoning applies for BERT-ADA Rest, which classifies "aren't audible" as negative.

Samples predicted incorrectly by the target-domain adapted model

In the following, we investigate examples that are classified incorrectly by the BERT-ADA models. This helps us to understand the remaining error types and shows a way forward for future work. The majority of incorrect predictions come from the ground-truth neutral class, which in most cases is confused with the positive class for restaurants and with the negative class for laptop reviews.
RE1 – influenced by sentiment towards a different aspect-target:
This example was classified correctly only by the BERT-ADA Laptop model. The reduced input for this model is the word *"'had"*, which is used a lot in fact based formulations like for example *"the CPU had 3 GHz"*. From experience, we think that this type of formulation appears more often in the laptops than in the restaurant domain. The BERT-ADA Restaurant and BERT-base model both seem to be influenced by the sentiment associated with another aspect-

target.
RE2 – influenced by sentiment towards a different aspect-target:
Words indicating a certain kind of relation to the aspect-target like "with" in this example could be used to separate the aspect-target specific sentiment from the general sentiment. We think that with more supervised data this case should be solvable by learning these relations in a general way.
LE1 – absence of something like a part classified as negative:
"not" is classified as negative by BERT-ADA Lapt. In the laptops domain the largest remaining confusion are neutral examples classified as negative examples by our algorithm. It seems like if absences of parts like a "disk drive" are mentioned, the algorithm tends to classify this as negative. In other examples these statements of absence of things actually imply a negative sentiment.
LE2 – possibly incorrect ground truth:
A handful of examples like this one are, in our opinion, labelled incorrectly. We think the word *"only"* indicates negative sentiment in this example.

Summary
To summarize, we find that in order to correctly predict aspect-target based sentiment, the context sensitivity of the sentiment expression plays an important role in difficult examples. By finetuning the language model on domain-specific text the model is able to capture this knowledge most of the time, even if such expressions are not directly observed in the training set used for downstream-classification.
We find that especially neutral examples are more difficult to classify correctly. Some of these examples could be solved for an applied real-world case with more supervised data that allows to learn more abstract relationships between entities like sauce and its ingredients in example RE2 and contain more fact-based formulations to discriminate the neutral class better. We also think that selecting finetuning corpora more carefully with these error types in mind could also lead to improvements of classification performance on these datasets.

5. Conclusion

We performed experiments on the task of Aspect-Target Sentiment Classification by first finetuning a pre-trained BERT model on a domain specific corpus with subsequent training on the down-stream classification task.
We analyzed the behavior of the number of domain-specific BERT language model finetuning steps in relation to the end-task performance.
With the findings on how to best exploit BERT language model finetuning we were able to train high performing models, of which the one trained on SemEval 2014 Task 4 restaurants dataset achieves new state-of-the-art performance.
We further evaluated our models cross-domain to explore the robustness of Aspect-Target Sentiment Classification. We found that with our setup, this task transfers well between the laptops and the restaurants domain.
As a special case we ran a cross-domain adaptation experiments, where the BERT language model is specifically finetuned on the target domain. We achieve significant improvement over unadapted models: one cross-domain adapted model performs even better than a BERT-base model that is trained in-domain.
Overall, our findings reveal promising directions for follow-up work. The XLNet-base model performs strongly on the ATSC task. Here, domain-specific finetuning could probably bring significant performance improvements. Another interesting direction for future work would be to investigate cross-domain behavior for an additional domain like hotels, which is more similar to the restaurants domain.

6. Bibliographical References

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). 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.

Feng, S., Wallace, E., Grissom II, A., Iyyer, M., Rodriguez, P., and Boyd-Graber, J. (2018). Pathologies of neural models make interpretations difficult. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3719–3728, Brussels, Belgium, October-November. Association for Computational Linguistics.

He, R. and McAuley, J. (2016). Ups and Downs. In *Proceedings of the 25th International Conference on World Wide Web - WWW '16*, pages 507–517, New York, New York, USA. ACM Press.

He, R., Lee, W. S., Ng, H. T., and Dahlmeier, D. (2018). Exploiting document knowledge for aspect-level sentiment classification. In *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, volume 2, pages 579–585.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. In *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, volume 1, pages 328–339.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '04*, page 168.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, Z., Wei, Y., Zhang, Y., Zhang, X., Li, X., and Yang, Q. (2019). Exploiting Coarse-to-Fine Task Transfer for Aspect-level Sentiment Classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4253—-4260.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep Con-

textualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., and Manandhar, S. (2015). SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Stroudsburg, PA, USA. Association for Computational Linguistics.

Radford, A. and Salimans, T. (2018). Improving Language Understanding by Generative Pre-Training. *URL https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf*.

Ruder, S. (2019). *Neural Transfer Learning for Natural Language Processing*. Ph.D. thesis.

Song, Y., Wang, J., Jiang, T., Liu, Z., and Rao, Y. (2019). Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314*.

Sun, C., Huang, L., and Qiu, X. (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.

Tang, D., Qin, B., Feng, X., and Liu, T. (2016a). Effective LSTMs for target-dependent sentiment classification. In *COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers*, pages 3298–3307.

Tang, D., Qin, B., and Liu, T. (2016b). Aspect Level Sentiment Classification with Deep Memory Network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-Decem, pages 5999–6009.

Xu, H., Liu, B., Shu, L., and Yu, P. S. (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. Association for Computational Linguistics.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Zhaoa, P., Houb, L., and Wua, O. (2019). Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification. *arXiv preprint arXiv:1906.04501*.