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.ai at Innsbruck, Austria
`{firstname.lastname}@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 the BERT language model, we approach ATSC by 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 enables 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 compared to strong baseline models like vanilla BERT-base and XLNet-base.

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 *positve, negative* or *neutral* sentiment polarity.

In constrast to SA, the more fine-grained task of Aspect Based Sentiment Analysis (ABSA) (Hu and Liu, 2004; Pontiki et al., 2015) aims at finding 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 is classified. For our example-sentence the correct result is *("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 to the aspect-target is determined. In our example the correct result is the tuple *("dumplings", "positive")*.

In this work, we focus on ATSC. In the last years, specialized neural architectures (Tang et al., 2016a,b) 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, including ABSA (Song et al., 2019; Xu et al., 2019; Sun et al., 2019). We still see huge potential that comes with this trend, this 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 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. The analysis of the influence of the amount of training-steps used for BERT language model finetuning on the Aspect-Target Sentiment Classification performance.

2. The findings on how to exploit BERT language model finetuning enables us to achieve new state-of-the-art performance on the SemEval 2014 restaurants dataset.

3. The analysis of cross-domain adaptation between the laptops and restaurants domain. Adaptation is tested by finetuning the BERT language model self-supervised 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.

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. Unfortunately, both datasets only have 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.

Knowledge Transfer for Aspect-Target Sentiment Classification Analysis

Another 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 the 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 finetuning 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). Finally, we discuss 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.

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 very 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 2 parts.

The first part is the masked language model objective, where the model learns to predict tokens, which have been randomly masked, from the 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 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 Mio. parameters in total.

For finetuning the BERT language model on a specific domain we use the weights of 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 shown 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 why domain-specific finetuning can 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 if 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 are occuring 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 $h_{[CLS]} \in \mathbf{R}^{768 \times 1}$. 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 $W \in \mathbf{R}^{3 \times 768}$. Cross-entropy is used as the training loss. The way we use BERT for classifying the sentiment polaritites 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*. Additionally, the model itself can be adapted towards a target domain. This 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 on 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 case the training datasets for laptops and restaurants are simply combined. The domain for testing D_{Test} can only be take on the values *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 best 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*.

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

RQ3: When 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 exploitet?

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 multiple marked aspect terms that each have a 3-level sentiment polarity (*positive, neutral or negative*) associated. In the original dataset the *conflict* label is also present. Here, conflicting labels are dropped for reasons of comparability with Xu et al. (2019). Both datasets are small, detailed statistics 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 less than two sentences.

For the laptop corpus, $1,007,209$ sentences are left after pre-processing. For the restaurants domain more reviews are available, we sampled $10,000,000$ sentences to have a sufficient amount of data for fully exploitet 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 1 sentence can be seen multiple times with a different language model masking.

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

To be able to reproduce our finetuning corpora, we make the code that is used to generate them available online[3].

Corpus	Sentences	Train 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.

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 Table 1, we finetune the language models on each domain so that the model trains a total of about 30 Mio. sentences (7.5 Mio. 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 to-

---

[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

tal 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 current 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 finetuning prior to downstream classification, where the BERT language model is finetuned jointly with a question answering task. It performs state-of-the-art 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 baselines: BERT and XLNet.

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 in BERT a more general permutation objective is used, where all possible variants of masking are fully exploitet.

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 laptops 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.
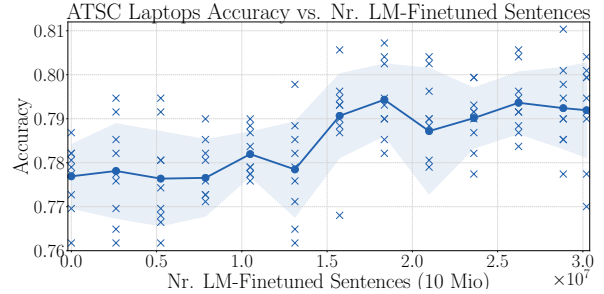

Figure 1: Accuracy of Aspect-Target Sentiment Classification as a function of the number of sentences the BERT language model has been finetuned on. Marked dots (•) connected through the line are the averages (μ) over 9 runs, a single run is marked as a cross (×). The standard deviation (σ) curves are also drawn ($\mu \pm \sigma$). The model is trained on the SemEval 2014 Task 4 laptops dataset. The language model is finetuned on our laptops domain corpus.

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 on domain-specific language model finetuning, we can see in Figure 1 that first of all, language model finetuning has a substantial effect on ATSC end-task performance. Secondly, we see that in the laptops domain the performance starts to increase at about 10 Mio. finetuned sentences. This is an interesting insight as one would expect a relation closer to a logarithmic curve. One reason might be that it takes many steps to train knowledge into the BERT language model due to its vast amount of parameters. The model already converges at around 17 Mio. sentences. More finetuning does not improve performance significantly. In addition, we find that different runs have a high variance, the standard deviation amounts to about 1% in accuracy, which justifies 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-

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.

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 finetuning 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 BERT$_{BASE}$. 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. Another finding is that BERT-ADA Joint performs better on the laptops dataset than BERT-ADA Rest, although the unique amount of laptop reviews are the same in laptops- and joint-corpora. We think that confusion can be created when mixing the domains, but this needs to be investigated further. We also find that the XLNet-base baseline performs generally stronger than BERT-base and even outperforms BERT-ADA Lapt with an accuracy of 79.89% on the laptops dataset.

To answer RQ3, which is concerned with domain adaptation, we can see in 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 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 many 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 leads to better performance if the domains do not confuse each other. Interesting for the joint-training case is that the BERT-ADA Joint model performs especially strong 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 label. 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.

5 Conclusion

We performed experiments on the task of Aspect-Target Sentiment Classification by first finetun-

ing a pre-trained BERT model on a domain specific corpus with subsequent training on the downstream 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, which one of even performs as new state-of-the-art on SemEval 2014 Task 4 restaurants dataset.

We further evaluated our models cross-domain to explore the robustness of Aspect-Target Sentiment Classification. We found that in general, 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, a 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. Here, it could be interesting to find out if the shared nature of these domain would results in more confusion or if they would behave synergetically.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 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.

Ruining He and Julian McAuley. 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.

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

Jeremy Howard and Sebastian Ruder. 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.

Minqing Hu and Bing Liu. 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.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Zheng Li, Ying Wei, Yu Zhang, Xiang Zhang, Xin Li, and Qiang Yang. 2019. Exploiting Coarse-to-Fine Task Transfer for Aspect-level Sentiment Classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4253—-4260.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized 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.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 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.

Alec Radford and Tim Salimans. 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*.

Sebastian Ruder. 2019. *Neural Transfer Learning for Natural Language Processing*. Ph.D. thesis.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314*.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for Aspect-Based Sentiment Analysis via Constructing Auxiliary Sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 380–385.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 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.

Duyu Tang, Bing Qin, and Ting Liu. 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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-Decem, pages 5999–6009.

Hu Xu, Bing Liu, Lei Shu, and Philip S 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. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Pinlong Zhaoa, Linlin Houb, and Ou Wua. 2019. Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification. *arXiv preprint arXiv:1906.04501*.