Unsupervised Data Augmentation for Aspect Based Sentiment Analysis

David Z. Chen **Adam Faulkner** **Sahil Badyal**
Capital One Servicing Intelligence, NLP
{david.chen2, adam.faulkner, sahil.badyal}
@capitalone.com

Abstract

Recent approaches to Aspect-based Sentiment Analysis (ABSA) take a co-extraction approach to this span-level classification task, performing the subtasks of aspect term extraction (ATE) and aspect sentiment classification (ASC) simultaneously. In this work, we build on recent progress in applying pre-training to this co-extraction task with the introduction of an adaptation of Unsupervised Data Augmentation (UDA) in semi-supervised learning. As originally implemented, UDA cannot accommodate span-level classification since it relies on advanced data augmentation techniques, such as backtranslation, that alter the sequence lengths of the original data and cause index mismatches. We introduce an adaptation of UDA using Masked Language Model (MLM) unmasking that accommodates this index-match constraint and test the approach on standard ABSA benchmark datasets. We show that simple augmentations applied to modest-sized datasets along with consistency training lead to competitive performance with the current ABSA state-of-the-art in the restaurant and laptop domains using only 75% of the training data.

1 Introduction

Aspect-based Sentiment Analysis (ABSA) is a subset of Sentiment Analysis (SA), operating at the phrase- rather then sentence- or document-level. As with other forms of SA, the goal is to determine the sentiment associated with a given text segment, though, in the case of ABSA, these phrasal segments are typically "aspects" or features associated with products, services, or experiences, such as "waitstaff" or "ambience."

As with other span-level classification tasks, such as Named Entity Recognition (NER), a major challenge of ABSA is class imbalance, as the majority of token labels typically refer to non-aspect terms (Luo et al., 2020) and the terms themselves are of inconsistent phrase-level categories. This introduces considerable variance in aspect term labels and makes it difficult for models to effectively generalize to example terms outside those explicitly shown in the training data.

1.1 Related work

In related span-level tasks, previous work has shown that a joint/collapsed approach to entity and sentiment co-extraction out-performs a pipelined approach (Mitchell et al., 2013; Zhang et al., 2015). A joint approach refers to assigning two sets of tags, term and polarity, to each example, and a collapsed approach collapses the term and polarity tags into one term-polarity tag for each token. While there are merits to both approaches, we adopt a collapsed approach as it requires a simpler classifier architecture.

Pre-training and pre-trained language models (LMs) have been shown to provide state-of-the-art performance on many tasks within NLP. Applying these approaches to the ABSA task, Li et al. (Li et al., 2019b) and Luo et al. (Luo et al., 2020) have achieved state-of-the-art performance on restaurant and laptop reviews using pre-trained LM's and LM's with post-training (PTR), respectively. These results highlight the benefit of leveraging unlabeled data (with pre-training and post-training).

In this work, we explore and push the limits of using unlabeled data for the ABSA task by incorporating data augmentation and consistency training on top of pre-trained and post-trained BERT. We adopt an unsupervised data augmentation (UDA) technique based in semi-supervised learning (SSL) from Xie et al. (Xie et al., 2020), initially developed

Proceedings of the 29th International Conference on Computational Linguistics, pages 6746–6751
October 12–17, 2022.

UDA method	Original Text	Augmented Text
Backtranslation	Our meal was so tasty but the waitstaff kept making rude remarks!	Food was delicious but the waiter spoke rudely.
MLM unmasking	Our <MASK> was so tasty but the waitstaff kept making rude remarks!	Our lunch was so tasty but the waitstaff kept making rude remarks!

Table (1) Example augmentation methods and texts. In the backtranslation case, "meal"-"food" and "waitstaff"-"waiter" are index mismatched after augmentation, whereas MLM unmasking preserves token indices.

for document- and sentence-level classification tasks, and adapt it to the span-level classification setting. In UDA, unlabeled data is passed to the model in streams of pairs, where one stream contains the original unmodified input example and the other stream contains augmented examples created by transforming the original input using data augmentation techniques. In the paper by Xie et al. (Xie et al., 2020), the authors apply data augmentation to images (e.g. filters and image transformations) as well as to sentence-level textual data via backtranslation. This backtranslation approach, while powerful in creating augmented examples that differ greatly from the original while retaining semantic meaning, results in index-mismatch issues when applied to span-level tasks such as ABSA. For this reason, we adapted the original UDA implementation to work for span-level ABSA by applying simple token replacements using masked-language model (MLM) unmasking (additional details provided in the Method section). Our adapted form of UDA-based data augmentation shows competitive performance with the ABSA state-of-the-art using only 75% of the original labeled training data and 30k additional unlabeled examples.

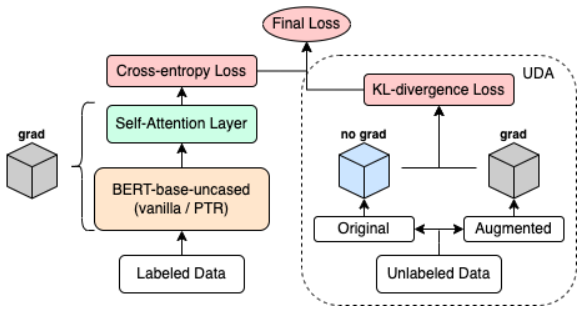

Figure (1) Model diagram with UDA

2 Method

Figure 1 shows the model architecture diagram for the models used in our experiments. One constant throughout is the use of *BERT-base-uncased* [1] as the base pre-trained LM and a Self-Attention Network (SAN) as the classification layer. In the purely supervised condition (no UDA), training is done in the usual way by calculating and backpropagating a cross-entropy loss between prediction and target, where collapsed labels are in the "BIOES" tagging scheme, and sentiment tags are appended to each BIES tag, e.g. B-POS, B-NEU, B-NEG, resulting in 13 classes in total.

2.1 Unsupervised Data Augmentation

For model variations using UDA, we include a separate model that performs backpropagation on unsupervised augmented datasets based on a KL-divergence loss between the model prediction on the augmented example (unfrozen) and the model prediction on the original example (frozen), as described in the original UDA paper (Xie et al., 2020). See Figure 1.

As currently implemented, advanced data augmentation techniques used in UDA, such as backtranslation, are incompatible with span-level classification tasks like ABSA, which require that the sequence length of the original example match that of the augmented example in order for the aspect terms to be correctly indexed, extracted and labeled. This is because backtranslation frequently results in augmented sequences of varying lengths from the original, leading to token index mismatches (see examples in Table 1). As a result, KL-divergence loss would fail to capture the error between the original and augmented aspects, as their relative positions will have changed.

In order to accomodate the index-match constraint required for ABSA, we introduce a simple augmentation technique that utilizes single- and multi-token replacement via unmasking using vanilla BERT-uncased MLM. This choice of BERT MLM is to remain consistent with previous work in ABSA utilizing BERT. Other choices can be used instead, such as one of the BERT variants, e.g. RoBERTa, DistilBERT (Liu et al., 2019; Sanh et al., 2019). In order to obtain augmentations that are general, we

[1] Available in *Hugging Face*: https://huggingface.co

Model		Rest14	Rest15	Rest16	Laptop
IMN	(He et al., 2019)	69.54	59.18	-	58.37
DREGCN	(Liang et al., 2020a)	72.60	62.37	-	63.04
WHW	(Peng et al., 2020)	71.95	65.79	71.73	62.34
TAS-BERT	(Wan et al., 2020)	-	66.11	75.68	-
IKTN-BERT	(Liang et al., 2020b)	71.75	62.33	-	62.34
DHGNN	(Liu et al., 2020)	68.91	58.37	-	59.61
RACL-BERT	(Chen and Qian, 2020)	75.42	66.05	-	63.40
BERT-E2E-ABSA	(Li et al., 2019b)	73.68	59.90	70.51	61.12
GRACE	(Luo et al., 2020)	77.26	68.16	76.49	**70.71**
UDA-ABSA		**79.38** ± 0.38	**70.14** ± 0.89	**78.05** ± 0.72	69.55 ± 0.40
-75% train sample		77.09 ± 0.52	68.19 ± 0.70	75.38 ± 0.27	-
-50% train sample		76.73 ± 0.17	64.02 ± 1.01	72.96 ± 0.70	-
-25% train sample		73.33 ± 0.24	58.71 ± 1.07	68.72 ± 1.45	-

Table (2) Experimental micro F1 values compared across previous work and UDA-ABSA

chose to utilize vanilla BERT-uncased for this MLM task rather than one post-trained on in-domain data.

Span-based UDA Single-unigram replacement augmentation

1: Randomly select a token in the tokenized original sequence (avoiding punctuation)
2: Convert the selected token into [MASK]
3: Unmask the token using our LM,
4: Check $token_{unmasked} \neq token_{original}$ and $token_{unmasked} \notin punctuation$.

In multi-unigram replacement augmentation, we iterate on the single-unigram case over S_{length} times, where S_{length} is the sequence length. However, we apply a confidence threshold, $\gamma = 0.1$, to the unmasking so that only unmasked tokens with confidence $> \gamma$ are kept.

2.2 Datasets

We leveraged datasets from two domains, namely Restaurants (Rest14, Rest15, Rest16) and Laptops, both originating from SemEval (Pontiki et al., 2014, 2015, 2016). Specifically, we use versions prepared by Li et al. (Li et al., 2019a), which uses collapsed ABSA labels. For UDA, we utilized the 27k examples from the Yelp academic dataset for the Restaurant domain, [2] and we filtered Amazon electronics reviews to obtain 38k examples pertaining to laptops. [3]

[2]https://www.yelp.com/academic_dataset
[3]http://jmcauley.ucsd.edu/data/amazon/

2.3 Model parameters

All of our models share the same underlying architecture: BERT-base-uncased (vanilla and post-trained) with a learning rate of $2e-5$ and an AdamW optimizer with a linear learning rate schedule. For post-training (PTR), we adopted the trained weights from Luo, et al. (Luo et al., 2020), which was achieved by performing Whole Word Masking using BERT-based-uncased on 142.8M Amazon reviews (footnote) joined with 2.2M Yelp reviews (footnote).

Experiments are conducted using a fixed number of gradient optimization steps, which is set heuristically and is sufficiently high to allow for complete model convergence (loss plateaus over at least one entire epoch). The actual number of steps varied depending on the batch sizes involved, but ranged from 5k-20k batches, or 8-15 epochs. The supervised batch sizes were set according to GPU memory limitations, as UDA greatly increases the GPU memory load during training, resulting in typical supervised batch sizes of 4 for Restaurant data, and 6 for Laptop data with UDA, and 16 and 32 for experiments without UDA, respectively. Likewise, the unsupervised batch sizes for UDA were set according to the ratio of $N_{unsupervised}/N_{supervised}$ multiplied by the supervised batch size, resulting in typical UDA batch sizes of 40-84.

Each experimental condition is conducted 5 times with 5 constant seeds, and the result-

ing micro-F1 values are averaged over those 5 replicates. We also provide the resulting standard deviations. This is performed in order to calculate statistical significance, as well as increase the confidence of our estimate of the average performance. For each iteration of the 5 total replicates, we estimate test micro-F1 by calculating a a 800-step moving average (∼1.2 epochs) on the test dataset after model convergence.

3 Results

3.1 Benchmark experiments

Table 2 shows the results of our experiments on the rest14, rest15, rest16, and laptop datasets. We found that UDA-ABSA is competitive with the state-of-the-art, and achieves this with pre-trained BERT-uncased, PTR, and UDA on additional unlabeled data during training with as little as 75% of the labeled data.

Model	**Rest14**
UDA-ABSA	**79.38** ± 0.38
no UDA	78.75 ± 0.38
no PTR	75.16 ± 0.33
no UDA & no PTR	73.50 ± 0.90

Table (3) UDA-ABSA Ablation Experiments

3.2 Ablation experiments

In order to assess the relative contributions of UDA and PTR to the model's performance, we conducted ablation experiments on the Rest14 dataset. Namely, we estimated model micro-F1 *w/o PTR* and *w/o UDA* and *w/o UDA and PTR*. Results are shown in Table 3. Removing PTR contributed a decrease of roughly ∼4% average micro F1, while removing UDA contributed a small, but significant (p val ≈ 0.048), decrease of $< 1\%$ average micro F1. Additionally, removing both PTR and UDA contributed a decrease of roughly ∼6% average micro F1. These observations suggest that:

- PTR contributes the most to enhancing the performance of the model.
- UDA may contribute more to the performance in the absence of PTR.
- While PTR leverages over 140M unlabeled examples, UDA improves model performance further.
- UDA may be more data-efficient: i.e. gain per number of ex. of PTR (∼0.03%/Million examples) vs. UDA (∼21%/Million examples).

However, it is not clear how such performance estimates would extrapolate to the small data regime for PTR and large data regime for UDA, as such experiments have not been conducted due to hardware constraints.

Model	**Rest14**
Single-Linear	**79.38** ± 0.38
Single-Log	79.19 ± 0.34
Single-Exp	79.10 ± 0.29
Multi-Linear	79.05 ± 0.37
Single-no CT	74.03 ± 0.32

Table (4) UDA Parameter Experiments

4 Discussion

In our experiments, we found that, not surprisingly, downsampling the training data results in a degradation of model performance, but we also found that UDA achieves SOTA-competitive performance with 75% of the data. The original UDA paper (Xie et al., 2020) showed good performance with only 20 training examples for sentiment classification. In our dataset, we achieved performance competitive with BERT-based-uncased (Li et al., 2019b) with only 25% of the training data.

In addition to benchmarks and ablation experiments, we explored the role of different types of confidence thresholding (CT) on the final performance of our models. The confidence threshold, ϕ, filters out unsupervised examples that fall below ϕ during UDA, so that the model does not reinforce its own errors, and ϕ is typically increased during training on a schedule (Xie et al., 2020), namely, *linear*, *log*, and *exp*.

We found that CT during UDA is important to the stability and convergence of our models. Figure 2 shows the training curves for test data micro-F1 with and without CT. Our models converge faster with CT, and interestingly, the curve for our models without CT shows a kink after ∼1 epoch, where micro-F1 appears to saturate only to rebound as training continues. We hypothesize that this kink is the result of the model initially learning the wrong features, which are subsequently relearned during further training. However, the final performance of our models without CT never reaches those with CT see Table 4 and Figure 2. This observation highlights the importance of CT for UDA and supports other recent work in SSL that have found success with such concepts (Sohn et al.,

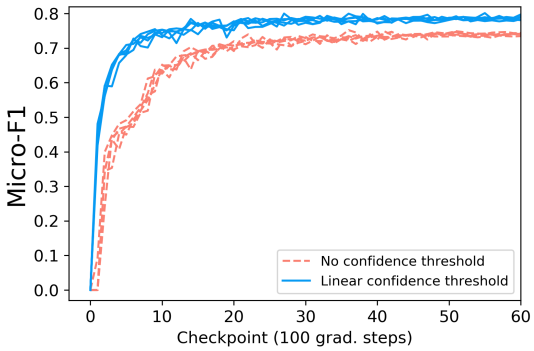

Figure (2) Rest14 test micro-F1 with and without confidence thresholding

2020).

5 Summary

We introduced a span-level modification of the UDA procedure, which, along with post-training on BERT, achieves performance competitive with state-of-the-art on the restaurant and laptop domains for ABSA with 75% of the data. While post-training contributed the most to overall performance, UDA may be more efficient on a per data/compute basis. We observed that confidence thresholding is essential to stabilize model training and achieve greater performance, and that linear confidence threshold scheduling achieved the best performance along with single augmentations compared to multiple augmentations. This work reveals the benefit of using UDA for span-level tasks and with post-trained language models.

References

Zhuang Chen and Tieyun Qian. 2020. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3694, Online. Association for Computational Linguistics.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2019. An interactive multi-task learning network for end-to-end aspect-based sentiment analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 504–515, Florence, Italy. Association for Computational Linguistics.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019a. A unified model for opinion target extraction and target sentiment prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6714–6721.

Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. 2019b. Exploiting BERT for end-to-end aspect-based sentiment analysis. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 34–41, Hong Kong, China. Association for Computational Linguistics.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2020a. A dependency syntactic knowledge augmented interactive architecture for end-to-end aspect-based sentiment analysis. *CoRR*, abs/2004.01951.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2020b. An iterative knowledge transfer network with routing for aspect-based sentiment analysis. *CoRR*, abs/2004.01935.

Shu Liu, Wei Li, Yunfang Wu, Qi Su, and Xu Sun. 2020. Jointly modeling aspect and sentiment with dynamic heterogeneous graph neural networks. *CoRR*, abs/2004.06427.

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. *ArXiv*, abs/1907.11692.

Huaishao Luo, Lei Ji, Tianrui Li, Daxin Jiang, and Nan Duan. 2020. GRACE: Gradient harmonized and cascaded labeling for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 54–64, Online. Association for Computational Linguistics.

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1643–1654, Seattle, Washington, USA. Association for Computational Linguistics.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8600–8607.

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.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In *5th Workshop on Energy Efficient Machine Learning and Cognitive Computing @ NeurIPS 2019*.

Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. 2020. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *34th Conference on Neural Information Processing Systems*. NeurIPS.

Hai Wan, Yufei Yang, Jianfeng Du, Yanan Liu, Kunxun Qi, and Jeff Z. Pan. 2020. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9122–9129.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. 2020. Unsupervised data augmentation for consistency training. In *34th Conference on Neural Information Processing Systems*. NeurIPS.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2015. Neural networks for open domain targeted sentiment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 612–621, Lisbon, Portugal. Association for Computational Linguistics.