

Exploiting Position Bias for Robust Aspect Sentiment Classification

Fang Ma*, Chen Zhang*, Dawei Song†

School of Computer Science, Beijing Institute of Technology
Beijing, China

{mfang, czhang, dwsong}@bit.edu.cn

Abstract

Aspect sentiment classification (ASC) aims at determining sentiments expressed towards different aspects in a sentence. While state-of-the-art ASC models have achieved remarkable performance, they are recently shown to suffer from the issue of robustness. Particularly in two common scenarios: when domains of test and training data are different (out-of-domain scenario) or test data is adversarially perturbed (adversarial scenario), ASC models may attend to irrelevant words and neglect opinion expressions that truly describe diverse aspects. To tackle the challenge, in this paper, we hypothesize that position bias (i.e., the words closer to a concerning aspect would carry a higher degree of importance) is crucial for building more robust ASC models by reducing the probability of mis-attending. Accordingly, we propose two mechanisms for capturing position bias, namely position-biased weight and position-biased dropout, which can be flexibly injected into existing models to enhance representations for classification. Experiments conducted on out-of-domain and adversarial datasets demonstrate that our proposed approaches largely improve the robustness and effectiveness of current models.¹

1 Introduction

Aspect sentiment classification (ASC) is an important sub-task of sentiment classification. It aims to identify the sentiment polarity (i.e., negative, neutral, or positive) of a specified aspect in a sentence. Take “*Great food but the service was bad.*” as an example. For aspects *food* and *service*, their corresponding sentiment polarities are *positive* and *negative*, respectively.

* Fang Ma and Chen Zhang contribute equally to this work. The order is determined alphabetically.

† Dawei Song is the corresponding author.

¹The code and preprocessed data are available at <https://github.com/BD-MF/POS4ASC>.

Scenario	Example	Pred./Lb.
I.D.	Great food but the <u>service</u> was bad !	neg./neg.
O.O.D.	The <u>battery</u> has never worked well .	pos./neg.
Adv.	<u>Awful</u> food but the <u>service</u> was great !	neg./pos.

Figure 1: An illustration of how an ASC model IAN (Ma et al., 2017) might fail. Gradient saliency maps (Simonyan et al., 2014) with respect to the embedding of each word in I.D., O.O.D., and Adv. scenarios, along with the model predictions (Pred.) and corresponding ground truth labels (Lb.), are provided. Underlined words are aspects.

A challenge in ASC is how to model semantic relations between aspect terms and their contexts, which requires an ASC model to be only sensitive to the sentiment words actually depicting the target aspect terms. Although previous ASC models (Tang et al., 2016b; Li et al., 2018; Zhang et al., 2019a; Xu et al., 2019; Wang et al., 2020; Tang et al., 2020) have achieved promising results by modeling complex interactions between aspects and contexts, these models have recently been shown to suffer from the lack of robustness (Xing et al., 2020). The issue is particularly severe in two scenarios: 1) out-of-domain (O.O.D.) scenario: ASC models that perform well on training data often fail to generalize to test data in another domain; 2) adversarial (Adv.) scenario: ASC models can be easily fooled by small adversarially perturbed inputs, e.g. synonymous word substituted ones. To our best knowledge, none of current ASC models have been targeted at alleviating the robustness issue in above-mentioned two scenarios.

To fill this gap, inspired by a recent finding that highlighting words close to a target aspect (termed as *position bias*) would boost in-domain (I.D.) effectiveness of a model (Zhang et al., 2019b), we hypothesize that such position bias is also crucial for a robust ASC model in O.O.D. and Adv. sce-

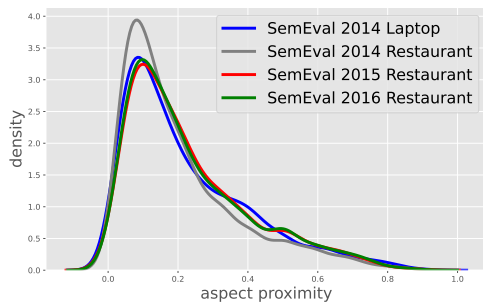


Figure 2: Aspect proximity distributions of different benchmarks, visualized with the kernel density estimation.

narios. Figure 1 shows an illustrative example of an ASC model that fails in the two scenarios due to mis-attending. In contrast, with position bias, a model tends to focus more on words nearer to the aspect, thus reducing the probability of mis-attending. Concretely, we propose two mechanisms: position-biased weight and position-biased dropout. The former assigns an inductive weight to each word according to its position proximity to the aspect. The latter gives each word a probability of being reserved (or dropped out) according to its proximity relation to the aspect. In doing so, position-biased weight degrades the significance of words that are not close enough to the aspect, while position-biased dropout will drop those likely irrelevant words at high probabilities.

Essentially, position bias is quantitatively evidenced in commonly used benchmarks. With annotated aspect-opinion pairs offered by Fan et al. (2019), we can calculate position proximity between any pair of aspect and opinion (in short, aspect proximity) in a sentence. The aspect proximity is computed by dividing the relative distance between a pair of aspect and opinion by the length of the corresponding sentence. Therefore, we can plot aspect proximity distributions of these benchmarks with kernel density estimation, as shown in Figure 2. These distributions indicate the aspect proximity is small at a high probability, thereby position bias is a reasonable inductive bias.

Extensive experiments are conducted on SemEval and ARTS datasets (Xing et al., 2020). The results show that incorporating the proposed position bias mechanisms would lead to more robust ASC models in both out-of-domain and adversarial scenarios. Furthermore, in terms of flexibility, the proposed methods can be easily adapted by subse-

quent models.

2 Capturing Position Bias

This section describes the proposed position-biased weight and position-biased dropout for capturing position bias. Formally, an n -word sentence containing a target m -word aspect term is formulated as $S = \{w_0, w_1, \dots, w_\gamma, w_{\gamma+1}, \dots, w_{n-1}\}$, where γ denotes the start index of the aspect term. By resorting to either a pre-trained word embedding (Bengio et al., 2003) or a pre-trained language model (Devlin et al., 2019), we can represent the sentence as $V = \{e_0, e_1, \dots, e_\gamma, e_{\gamma+1}, \dots, e_{n-1}\}$. We can then use the position-biased weight and dropout to refine V and generate an enhanced representation, denoted as $E = \{h_0, h_1, \dots, h_\gamma, h_{\gamma+1}, \dots, h_{n-1}\}$. E can then be incorporated into the model, i.e., any further structures for the model will be built upon E , instead of V , to predict sentiment polarities associated with diverse aspects.

Position-biased Weight Generally, the sentiment polarity of an aspect term is determined by its context, which are the words around the aspect term (Zhang et al., 2019b). Thus we can leverage relative position information to calculate weights of context words, with the aim to degrade the significance of those words that are far away from the aspect. Position-biased weight, denoted as $p_i \in (0, 1)$, is computed as:

$$p_i = \begin{cases} 1 - \frac{\gamma-i}{n-m} & 0 \leq i < \gamma \\ \frac{1}{n-m} & \gamma \leq i < \gamma + m \\ 1 - \frac{i-\gamma-m+1}{n-m} & \gamma + m \leq i < n \end{cases} \quad (1)$$

Then we compute h_i as: $h_i = p_i \cdot e_i$.

Position-biased Dropout Dropout (Srivastava et al., 2014; Sennrich et al., 2016) randomly sets elements in a feature vector to zeros. The word-level dropout can model semantic and syntactic compositionality and reduce input redundancy (Iyyer et al., 2015). Motivated by this idea, we give each word a probability of being reserved according to its position proximity to the aspect. The aim is to preserve those words that are close enough to the aspect and drop the rest. The probability that the i -th word will be preserved can be computed as:

$$z_i \sim \text{Bernoulli}(p_i) \quad (2)$$

where $\text{Bernoulli}(p_i)$ denotes that z_i equals to 1 with p_i and equals to 0 with $1 - p_i$. The i -th word is

dropped out if z_i is 0. Likewise, h_i can be attained by multiplying z_i and e_i .

3 Experiments

Datasets To empirically evaluate a model’s O.O.D. robustness, we conduct experiments on two domains from SemEval 2014 (Pontiki et al., 2014) dataset (SEMEVAL): one is laptop (SEMEVAL-LAP) and the other is restaurant (SEMEVAL-REST). For Adv. robustness, we experiment with the Aspect Robustness Test dataset (ARTS) (Xing et al., 2020), which is derived from the SemEval 2014 dataset. Instances in ARTS are generated with three adversarial strategies. These strategies enrich the test set from 638 to 1,877 for the laptop domain (ARTS-LAP), and from 1,120 to 3,530 for the restaurant domain (ARTS-REST). Note that each domain from SEMEVAL consists of separate training and test sets, while each domain from ARTS only contains a test set. Since SEMEVAL dataset does not come with development sets, 150 instances from the training set in each domain are randomly selected to form the development set. Table 1 shows the statistics of the datasets.

Dataset		# pos.	# neu.	# neg.
SEMEVAL-LAP	train	930	433	800
	test	341	169	128
	dev	57	27	66
SEMEVAL-REST	train	2,094	579	779
	test	728	196	196
	dev	70	54	26
ARTS-LAP	test	883	407	587
ARTS-REST	test	1,953	473	1,104

Table 1: Statistics of datasets.

Target Models We conduct experiments on a wide range of existing models for a comprehensive study on whether position bias is beneficial. Specifically, we examine these models’ performance before and after injecting the position bias, in terms of position-biased weight (`pos-wt`) and dropout (`pos-dp`) individually.

The target models include: (a) **LSTM** (Tang et al., 2016a) uses the last hidden state vector of the LSTM to predict sentiment. (b) **LSTM-Attn** (Wang et al., 2016) applies an attention-based LSTM on the concatenation of the aspect and word embeddings. (c) **IAN** (Ma et al., 2017) interactively learns attentions between context words and

aspect terms. (d) **MemNet** (Tang et al., 2016b) applies attention multiple times on word memories, and the output of the last attention is used for prediction. While the original work utilizes word embeddings as memories, we instead choose to add a layer of bidirectional LSTM upon embeddings for more abstractive memories. (e) **AOA** (Huang et al., 2018) introduces an attention-over-attention based network to model interaction between aspects and contexts. (f) **RoBERTa** (Dai et al., 2021) is a strong baseline with an MLP built upon the pooled feature induced with RoBERTa (Liu et al., 2019).

Implementation Details In all our experiments, the 300-dimensional GloVe (Pennington et al., 2014) is leveraged to initialize the input embedding. All parameters of models are initialized with uniform distributions. During all experiments, a bidirectional LSTM is adopted if necessary instead of a unidirectional one. If a model takes advantage of the attention mechanism, then dot product based attention is employed. In case a model has hidden states, the dimensionality of hidden states is set to 300. The batch size is 64. We use Adam (Kingma and Ba, 2015) as the optimizer with a learning rate of 10^{-3} . The coefficient of L2 regularization is 10^{-5} . For experiments with RoBERTa as the input embedding (Liu et al., 2019), things may change. The dimensionality of hidden states is 768. The learning rate is 10^{-5} , while the regularization is rather removed.

Evaluation Metrics For O.O.D., models are trained separately on one domain and evaluated on another. For Adv., models are trained on the SEMEVAL dataset and tested on the ARTS counterpart. For every test, a model is trained on the I.D. training set, selected on the I.D. development set, and tested on the O.O.D. or Adv. test set. The experimental results are obtained by averaging 5 runs with random initialization, and we adopt Accuracy and macro-averaged F1 scores as evaluation metrics.

In-domain Generalization Results Table 3 shows the I.D. performance of the LSTM on both laptop and restaurant domains, which exhibits incorporating position bias does not harm, if this is the case, a model’s generalization on I.D. test sets that much. On the contrary, position bias, especially with position-biased weight, can boost I.D. performance.

Model	LAP				REST			
	O.O.D.		Adv.		O.O.D.		Adv.	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
LSTM	71.02	52.15	49.49	43.91	60.60	53.25	53.34	41.99
w/ pos-dp	71.48↑0.46	50.98↓1.17	50.74↑1.25	44.38↑0.47	63.39↑2.79	58.57↑5.32	53.57↑0.23	42.11↑0.12
w/ pos-wt	72.96↑1.94	55.88↑3.73	55.50↑6.01	50.03↑6.12	66.33↑5.73	60.21↑6.96	59.03↑5.69	48.20↑6.21
LSTM-Attn	71.61	53.61	51.33	46.11	62.85	54.97	58.45	49.65
w/ pos-dp	71.34↓0.27	52.49↓1.12	53.76↑2.43	48.47↑2.36	65.24↑2.39	59.07↑4.10	58.64↑0.19	47.22↓2.43
w/ pos-wt	72.84↑1.23	56.18↑2.57	58.53↑7.20	53.54↑7.43	68.90↑6.05	64.48↑9.51	64.80↑6.35	55.34↑5.69
IAN	72.09	54.44	52.91	47.54	63.82	55.20	57.75	48.12
w/ pos-dp	70.95↓1.14	51.63↓3.08	52.04↓0.87	45.87↓1.67	63.57↓0.25	56.81↑1.61	56.89↓0.86	46.90↓1.22
w/ pos-wt	72.86↑0.77	54.88↑0.44	56.03↑3.12	50.30↑2.76	62.45↓1.37	55.95↑0.75	63.49↑5.74	54.04↑5.92
MemNet	70.66	52.07	52.00	46.50	57.84	51.15	55.30	46.67
w/ pos-dp	69.93↓0.73	53.37↑1.30	53.54↑1.54	47.93↑1.43	61.94↑4.10	54.49↑3.34	57.31↑2.01	45.23↓1.44
w/ pos-wt	70.67↑0.01	54.14↑2.07	56.04↑4.04	49.64↑3.14	61.35↑3.51	54.85↑3.70	61.10↑5.80	51.49↑4.82
AOA	71.63	52.65	52.16	46.78	63.73	57.00	58.19	49.02
w/ pos-dp	72.30↑0.67	53.73↑1.08	53.56↑1.40	48.18↑1.40	65.33↑1.60	58.31↑1.31	56.24↓1.95	45.63↓3.39
w/ pos-wt	72.61↑0.98	56.54↑3.89	59.07↑6.91	54.92↑8.14	66.87↑3.14	62.02↑5.02	64.35↑6.16	54.62↑5.60
RoBERTa	83.16	72.99	73.57	69.26	77.62	71.34	79.08	71.79
w/ pos-dp	81.98↓1.18	70.81↓2.18	69.98↓3.59	65.35↓3.91	75.61↓2.01	68.00↓3.34	77.81↓1.27	69.37↓2.42
w/ pos-wt	83.43↑0.27	74.08↑1.09	75.72↑2.15	72.09↑2.83	79.40↑1.78	74.44↑3.10	79.47↑0.39	73.10↑1.31

Table 2: Robustness results (%). O.O.D. on LAP or REST denotes a model is trained in current domain (LAP or REST) and tested on another (REST or LAP). Adv. denotes a model is trained in a domain and tested on its ARTS counterpart. Furthermore, w/ pos-dp means a model with position-biased dropout. w/ pos-wt means a model with position-biased weight. The small number next to each performance score indicates either performance improvement (↑) or drop (↓) compared with the original model without using position bias, and those highlighted in red are the best-performing ones among two variants.

Model	LAP I.D.		REST I.D.	
	Acc.	F1	Acc.	F1
LSTM	67.15	60.57	74.57	62.14
w/ pos-dp	67.34	60.27	74.23	61.55
w/ pos-wt	68.78	62.42	76.34	64.85

Table 3: I.D. results (%) of LSTM on LAP and REST.

Robustness Results The robustness results are shown in Table 2. We can see that performance of LSTM drops drastically, compared to I.D. performance, on O.O.D. and Adv. test sets, indicating the importance of studying the robustness issue. Our proposed two position bias mechanisms improve the target models’ O.O.D. and Adv. performance in most cases. With position-biased dropout, F1 scores of models are improved by up to 5.32 pp on O.O.D. test sets, and 2.36 pp on Adv. test sets, though the efficacy of the position-biased dropout seems not stable across different target models and settings. In contrast, the impact of position-biased weight is much more prominent. With position-biased weight, Accuracy scores of models can be

enhanced by up to 6.05 pp and 7.20 pp on O.O.D. and Adv. test sets, respectively. Further, F1 scores of models are improved by up to 9.51 pp and 8.14 pp on O.O.D. and Adv. test sets.

A highlight is that experimental results with RoBERTa as well exhibit the benefit of position bias, yet with caveats. Although pre-trained language models like RoBERTa are subject to positional encodings, such absolute position information is not enough to model relative position relations between aspect terms and contexts. Therefore, position bias matters during fine-tuning pre-trained language models for robust ASC performance. However, we observe that position-biased dropout is not an appropriate choice for pre-trained language models.

Case Study To understand the effect of position bias, we conduct a case study on the two robustness scenarios, as shown in Table 4. Specifically, we visualize the attention scores separately offered by an ASC model LSTM-Attn with and without position-biased weight method and trained on SEMEVAL-REST.

We can observe that before applying position bias, the model attends irrelevant words and fails in both scenarios. Specifically, in both cases, the model mis-attends to irrelevant opinion expressions. After injecting position bias, the attention scores become more accurate and the model attends to the correct opinion spans.

w/ pos-wt	Example
✗	The <u>price</u> is reasonable although the quality is poor .
✓	The <u>price</u> is reasonable although the quality is poor .
✗	Awful food but the <u>service</u> was great !
✓	Awful food but the <u>service</u> was great !

Table 4: **Case study.** The underlined words are aspects. The top two rows are O.O.D. examples, while the bottom two are Adv. examples. ✗ and ✓ refers to without and with pos-wt respectively.

4 Related Work

Fine-grained Sentiment Analysis ASC falls in the broad scope of fine-grained sentiment analysis. While ASC is basically formulated as determining sentiment polarity of a given aspect in a sentence (Tang et al., 2016b,a; Wang et al., 2016; Chen et al., 2017; Huang et al., 2018; Li et al., 2018; Xu et al., 2019; Zhang et al., 2019b,a; Wang et al., 2020; Tang et al., 2020), there is an emergent trend that treating fine-grained sentiment as an opinion triplet extraction task (Peng et al., 2020; Zhang et al., 2020; Wu et al., 2020). Recently, the robustness of ASC models becomes a critical issue that urges researchers to pay more attention on improving the robustness of ASC models (Xing et al., 2020). Our work is the first work to enhance the universal robustness of ASC models by capturing position bias. On another note, we believe opinion triplet extraction is exposed to the similar robustness issue, which should be explored in the near future.

Robustness in NLP Broadly, there are two kinds of robustness in NLP, i.e., O.O.D. and Adv. robustness. O.O.D. robustness in NLP has attracted extensive attention in recent work (Ng et al., 2020; Hendrycks et al., 2020; Xie et al., 2020). In terms of O.O.D. robustness, they often use the cross-domain setting to evaluate models (Benson and

Ecker, 2020). Previous work mainly focuses on how to minimize the domain discrepancy and how to improve the feature adaptability of models (Rietzler et al., 2020; Ye et al., 2020). On the other hand, adversarial learning becomes the main method used to improve Adv. robustness of models (Xing et al., 2020). Prior methods consider using semantic operations, such as synonym replacement, random insertion, random swap, and random deletion to augment data (Wei and Zou, 2019). Other methods involve adding extra text (Wallace et al., 2019) and replacing sentences with semantically similar sentences (Ribeiro et al., 2018). Our work goes beyond the two forms of robustness and aims to achieve universal robustness for ASC with position bias.

5 Conclusion and Future Work

In this work, we find that state-of-the-art ASC models suffer from the issue of robustness, particularly in two scenarios: i) out-of-domain scenario, and ii) adversarial scenario. To address the issue, we propose a simple yet effective inductive bias that should be incorporated, that is, position bias. We proposed two mechanisms to capture position bias, namely position-biased weight and position-biased dropout. They are injected into existing models to enhance the representation. Extensive experiments demonstrate that the proposed methods can largely improve the models’ robustness. The results verify our hypothesis that position bias is beneficial for building more robust ASC models.

The work shall be improved in the following two facets: i) Since the approach of incorporating position bias is straightforward yet naive, especially for pre-trained language models, it is meaningful to consider a nicely designed architecture to inject position bias in a more elegant manner. ii) It has been shown that position bias for ASC is highly correlated with the syntactic structure of the sentence. Hence, syntax can likewise be explored to enhance the robustness of ASC models.

Acknowledgements

This work is supported by the National Key Research and Development Program of China (grant No. 2018YFC0831704) and the Natural Science Foundation of China (grant No. U1636203).

References

- Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *The journal of machine learning research*, 3:1137–1155.
- Vitus Benson and Alexander Ecker. 2020. Assessing out-of-domain generalization for robust building damage detection. *arXiv preprint arXiv:2011.10328*.
- Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 452–461.
- Junqi Dai, Hang Yan, Tianxiang Sun, Pengfei Liu, and Xipeng Qiu. 2021. Does syntax matter? a strong baseline for aspect-based sentiment analysis with roberta. *arXiv preprint arXiv:2104.04986*.
- 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.
- Zhifang Fan, Zhen Wu, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. 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 2509–2518.
- Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedziec, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2744–2751.
- Binxuan Huang, Yanglan Ou, and Kathleen M Carley. 2018. Aspect level sentiment classification with attention-over-attention neural networks. In *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, pages 197–206.
- Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1681–1691.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*.
- Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 946–956.
- 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 preprint arXiv:1907.11692*.
- Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4068–4074.
- Nathan Ng, Kyunghyun Cho, and Marzyeh Ghassemi. 2020. Ssmba: Self-supervised manifold based data augmentation for improving out-of-domain robustness. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1268–1283.
- 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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8600–8607.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.
- Maria Pontiki, Haris Papageorgiou, Dimitrios Galanis, Ion Androutsopoulos, John Pavlopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. *SemEval 2014*, page 27.
- Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging nlp models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865.
- Alexander Rietzler, Sebastian Stabinger, Paul Opitz, and Stefan Engl. 2020. Adapt or get left behind: Domain adaptation through bert language model fine-tuning for aspect-target sentiment classification. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4933–4941.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Edinburgh neural machine translation systems for wmt 16. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 371–376.

- Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *ICLR*.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.
- Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective lstms for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: 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.
- Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6578–6588.
- Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. 2019. Universal adversarial triggers for attacking and analyzing nlp. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.
- Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238.
- Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615.
- Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6383–6389.
- Zhen Wu, Chengcan Ying, Fei Zhao, Zhifang Fan, Xinyu Dai, and Rui Xia. 2020. Grid tagging scheme for end-to-end fine-grained opinion extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2576–2585.
- Sang Michael Xie, Ananya Kumar, Robbie Jones, Fereshte Khani, Tengyu Ma, and Percy Liang. 2020. In-n-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. *arXiv preprint arXiv:2012.04550*.
- Xiaoyu Xing, Zhijing Jin, Di Jin, Bingning Wang, Qi Zhang, and Xuan-Jing Huang. 2020. Tasty burgers, soggy fries: Probing aspect robustness in aspect-based sentiment analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3594–3605.
- Hu Xu, Bing Liu, Lei Shu, and S Yu Philip. 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.
- Hai Ye, Qingyu Tan, Ruidan He, Juntao Li, Hwee Tou Ng, and Lidong Bing. 2020. Feature adaptation of pre-trained language models across languages and domains with robust self-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7386–7399.
- Chen Zhang, Qiuchi Li, and Dawei Song. 2019a. Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4560–4570.
- Chen Zhang, Qiuchi Li, and Dawei Song. 2019b. Syntax-aware aspect-level sentiment classification with proximity-weighted convolution network. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1145–1148.
- Chen Zhang, Qiuchi Li, Dawei Song, and Benyou Wang. 2020. A multi-task learning framework for opinion triplet extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 819–828.