# ABSA-Bench: Towards the Unified Evaluation of Aspect-based Sentiment Analysis Research

**Abhishek Das**
School of Computer Science
The University of Adelaide
abhishek.das@student.adelaide.edu.au

**Wei Emma Zhang**
School of Computer Science
The University of Adelaide
wei.e.zhang@adelaide.edu.au

## Abstract

Aspect-Based Sentiment Analysis (ABSA) has gained much attention in recent years. ABSA is the task of identifying fine-grained opinion polarity towards a specific aspect associated with a given target. However, there is a lack of benchmarking platform to provide a unified environment under consistent evaluation criteria for ABSA, resulting in the difficulties for fair comparisons. In this work, we address this issue and define a benchmark, ABSA-Bench[1], by unifying the evaluation protocols and the pre-processed public datasets in a Web-based platform. ABSA-Bench provides two means of evaluations for participants to submit their predictions or models for online evaluation. Performances are ranked in the leader board and a discussion forum is supported to serve as a collaborative platform for academics and researchers to discuss queries.

## 1 Introduction

Aspect-based sentiment analysis (ABSA) has gained a lot of attention in recent years from both industries and academic communities as it provides a more practical solution to real life problems. The goal of ABSA is to identify the aspects and infer the sentiment expressed for each aspect. For example, given a sentence *I hated their service, but their food was great*, the sentiment polarities for the aspect *service* and *food* are negative and positive respectively. Conventional techniques for ABSA are mostly traditional machine learning models based on lexicons and syntactic features (Jiang et al., 2011; Kiritchenko et al., 2014; Vo and Zhang, 2015). Therefore, the performance of such models depend on hand-crafted features. Recent progresses have been made with the advancement of Deep Neural Networks (DNN) with some of the models being considered as state-of-the-art (Xu et al.). Among them, attention mechanism has played an important role outperforming previous approaches by paying more attention to the context words that are semantically-closer with the aspect terms (Luong et al., 2015; Wang et al., 2016; Chen et al., 2017; Liu et al., 2018; Ma et al., 2017). The most recent approaches adopted pre-trained Bidirectional Encoder Representations from Transformers (BERT) architecture (Devlin et al., 2019; Xu et al.) generating significant performance gaps to other approaches due to BERT's capability of capturing bi-directional contextual information and providing rich token-wise representation. Introducing BERT architecture into ABSA task naturally distinguishing the approaches to Non-BERT based models and BERT-based models. Figure 1 depicts the general processes of both of the two groups of supervised ABSA methods.

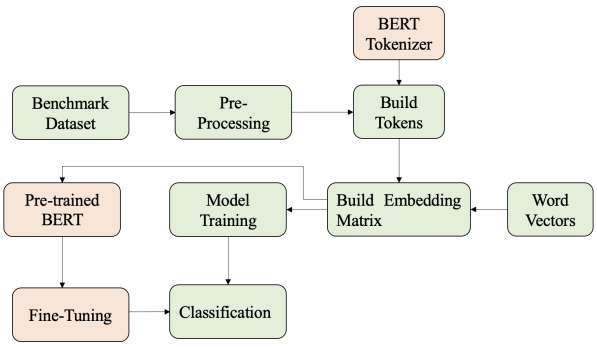


Figure 1: The General Process of ABSA

Although this research area has gained much attention in recent years, it lacks of unbiased comparisons overall. As deep learning based models perform differently on various hardware on different deep learning tools, existing works typically chose to either re-run or re-implement the selected comparative models under their own experimental environment. We also observe few works directly referring the results presented in the corresponding

[1] https://absa-bench.com/

papers for comparison. This makes it difficult to have a general overview of the performances of the state-of-the-art models and has motivated us to build a benchmarking platform for ABSA research.

Existing benchmarking research works are mostly conducted on evaluating single tasks and none of them support aspect-based sentiment analysis (Rajpurkar et al., 2016, 2018; Choi et al., 2018; Wang et al., 2019; Aguilar et al., 2020; Zhu et al., 2018). In this project, we fill this gap by proposing a unified evaluation process and building a united platform for comparing different ABSA models. We name our work as ABSA-Bench. ABSA-Bench particularly focuses on supervised approaches and is suitable for both DNN-based and conventional models. It provides two means of evaluations namely, Results Evaluation and Model Evaluation. Results evaluation is done by comparing the ground-truth with the model-generated predictions submitted by the researchers. Model evaluation supports the model submission and online evaluation which keeps the integrity of the predictions in a better way. To aid the model evaluation, a Web based tool is developed to provide an objective evaluation environment. The background computation power of ABSA-Bench is supported by the Google Cloud Platform (GCP)[2]. After evaluation, the performance results are then ranked in the ABSA-Bench leader board. ABSA-Bench further supports a discussion forum for queries, comments and discussions regarding the model implementations, performances, ranking and new ideas.

To the best of our knowledge, this is the first platform created with diverse functionalities to support the understanding of the state-of-the-art ABSA works. The contributions of the work includes: i) providing a unified ABSA evaluation platform which enables researchers to evaluate their models on the same benchmark dataset with a consistent metric under the same computation environment; ii) supporting a leader board for easy comparison, and a discussion forum for sharing ideas; iii) presenting the comparisons of several recent research works based on their performances on the ABSA-Bench platform through a re-run or re-implementation.

## 2 Related Works

The related benchmarking platforms for natural language processing models can be categorized into two groups: single task benchmarks and multiple tasks benchmarks. SQuAD (Rajpurkar et al., 2016, 2018)[3] is a representative benchmark for a single task. It provides a platform for evaluating question answering models on the SQuAD dataset. Researchers could either submit the prediction results or their models which will be run on CodaLab Worksheets[4]. A leader board ranks the performances of all the evaluated models. QuAC (Choi et al., 2018)[5] imitates SQuAD, but for context-aware question answering models for which the questions and answers are provided in the dialogue form. GLUE (Wang et al., 2019)[6] provides a collection of tools for evaluating the natural language understanding models across a diverse set of existing tasks. It allows researchers to submit their prediction files for comparison. Error analysis is also enabled. LinCE (Aguilar et al., 2020)[7] is a centralized benchmark for linguistic code-switching evaluation that combines ten corpora covering four different code-switched language pairs and four sub-tasks. Similar to GLUE, LinCE enables result submission, but does not support online model execution. TextGen (Zhu et al., 2018) is a benchmarking platform to support research on open-domain text generation models. It implements a majority of text generation models and aims to standardize the research in this field. However, TextGen does not allow online submission and evaluation.

ABSA-Bench is the most akin to SQuAD but unlike SQuAD, it focuses on ABSA task. ABSA-Bench provides two means of evaluations that is similar to SQuAD and QuAC. The online evaluation in ABSA-Bench is supported by JupyterHub which has key features like customization, flexibility and scalability. This distinguishes it from other similar platforms. JuptyerHub also serves a variety of environments. It can be easily containerised with any container, therefore can be scaled up for a greater number of users. A number of authentication protocols such as OAuth and GitHub are also supported, making it flexible for users. ABSA-Bench also supports an online discussion forum for researchers to exchange their ideas.

There are relatively less research efforts on providing a comprehensive benchmarking platform for multiple NLP tasks. DecaNLP (McCann et al.,

[2]https://cloud.google.com/

[3]https://rajpurkar.github.io/SQuAD-explorer/

[4]https://worksheets.codalab.org/

[5]http://quac.ai/

[6]https://gluebenchmark.com/

[7]https://ritual.uh.edu/lince/home

2018)[8] is the only one found in this category. It spans ten NLP tasks and recasts these tasks as question answering over a context using automatic transformations. Therefore, DecaNLP evaluates the models under the rubrics of assessing question answering models. DecaNLP considers the general sentiment analysis, but does not include ABSA.

## 3 Taxonomy and the Models

Aspect based sentiment analysis is a fundamental task in sentiment analysis research field (Pontiki et al., 2014) which comprises of three sub-tasks: aspect extraction, sentiment extraction and aspect based sentiment classification. In recent years, deep neural network has gained a lot of attention in solving the problem of ABSA. More recently, BERT (Devlin et al., 2019), has shown its effectiveness to alleviate the effort of feature engineering and has shown state-of-the art results in the given task. However these performance improvements have been achieved at a high computational cost. As a result these models are costly to train and evaluate. To have a better understanding of the large number of DNN based ABSA models, a categorization is utmost essential. Therefore, a taxonomy has been designed in this study which categorises different deep learning supervised technique, diving all approaches into broadly two categories: BERT based and Non-BERT based models. Note that we focus on supervised approaches in this work.

### 3.1 Models

Although the platform is designed for researchers to evaluate their models per their own need, we examined some representative models as examples.

#### 3.1.1 Non-BERT based Models

**CNN.** We adopt a **C**onvolution **N**eural **N**etwork model (Xue and Li, 2018) based on convolution operations and gating mechanisms to represent the CNN-based ABSA models.

**LSTM.** A vanilla **L**ong **S**hort **T**erm **M**emory network represents the vanilla RNN-based models.

**TD-LSTM. T**arget-**D**ependent **LSTM** (Tang et al., 2016a) is a modified LSTM. It consists of two LSTMs, which models the preceding and subsequent contexts surrounding the target words (aspect terms) respectively so that the contexts in both directions can be used as the feature representations for classifying sentiment in later stage.

[8] https://decanlp.com/

**TC-LSTM. T**arget-**C**onnection **LSTM** (Tang et al., 2016a) extends TD-LSTM by adding target connection component in order to capture the interactions between target word and its contexts. This component is basically a concatenation of word embedding and target vector at each position.

**ATAE-LSTM.** The **AT**tention-based LSTM with **A**spect **E**mbedding (Wang et al., 2016) model appends the aspect embedding into each word input vector to capture aspect information. To capture the inter-aspect dependencies, the aspect-focused sentence representations are fed into another LSTM to model the temporal dependency.

**CABASC. C**ontent **A**ttention **B**ased **A**spect based **S**entiment **C**lassification model (Liu et al., 2018) improves the attention mechanism with the help of two attention enhancing mechanisms, i.e., sentence-level content attention and context attention. This ensures that the model is capable of taking the word order information, the aspect information and the correlation between the word and the aspect to calculate the attention weight and embed them into a series of customized memories.

**IAN. I**nteractive **A**ttention **N**etwork considers attention mechanisms on both the aspect and the context (Ma et al., 2017). It uses two attention-based LSTM which interactively capture the key aspect terms and the important words of its context. The final representation of the sentence is produced by concatenating the representations of the aspect and its context, and is then passed to a soft-max layer for sentiment classification.

**MemNet.** A **Mem**ory **Net**work-based model (Tang et al., 2016b) adopts an attention mechanism with multi-hop layers which are stacked to select abstractive evidences from an external memory.

**RAM.** The **R**ecurrent **A**ttention mechanism based on **M**emory network (Chen et al., 2017) targets the cases that aspect terms are distant from the corresponding sentiment information. RAM introduces multiple attentions to distill the related information from its position-weighted memory and a recurrent network for sentiment classification.

#### 3.1.2 BERT based Models

**BERT-SPC.** In this model, a pre-trained BERT model was fine-tuned with just one additional layer (Devlin et al., 2019). For down-stream task like ABSA, the input representation is able to represent both a single sentence and a pair of sentences.

**AEN-BERT.** The **A**ttentional **En**coder **N**etwork (Song et al., 2019) is built upon a BERT embedding layer along with an attentional encoder layer and a target-specific attention layer.

**LCF-BERT.** In this model (Zeng et al., 2019), a Local Context Focus (LCF) mechanism is proposed for aspect-based sentiment classification based on multi-head self-attention . It utilizes the Context Features Dynamic Mask and Context Features Dynamic Weighted layers to assign more attention weights to the local context words. A BERT-shared layer is adopted to capture the internal long-term dependencies of local context and global context.

**BERT-PT.** The BERT **P**ost-**T**raining (Xu et al.) work enhances the performance of fine-tuning of BERT for Review Reading Comprehension (RRC) by adding a post-training step. This approach was then generalised to perform the task of aspect extraction and aspect sentiment classification in aspect-based sentiment analysis.

## 4 The ABSA-Bench

This section introduces the ABSA-Bench platform, including the two ways of ABSA benchmarking evaluations provided and our insights into the design and implementation of ABSA-Bench.

### 4.1 Evaluating the Results

To evaluate the model's performance, we provide a way for researchers to submit their prediction results on the formatted test set to ABSA-Bench. The submission file needs to follow the structure required by ABSA-Bench, which is simply the sentence ID and aspect terms along with the predicted sentiment polarity. We also make available an evaluation script that we will use for the official evaluations. The evaluation script will measure the model performance based on Macro $F1$ score, which is the weighted average of Precision and Recall. It is usually a more useful accuracy measure when there is an uneven class distribution which was the case in our benchmarking dataset.

### 4.2 Evaluating the Models

The other means of evaluation supported by ABSA-Bench is model evaluation. We provide a unified online computation environment for researchers to train and test their models. We used widely-adopted JuypterHub[9] to which researchers could submit their model as a Juypter Notebook file.

[9]https://jupyter.org/hub

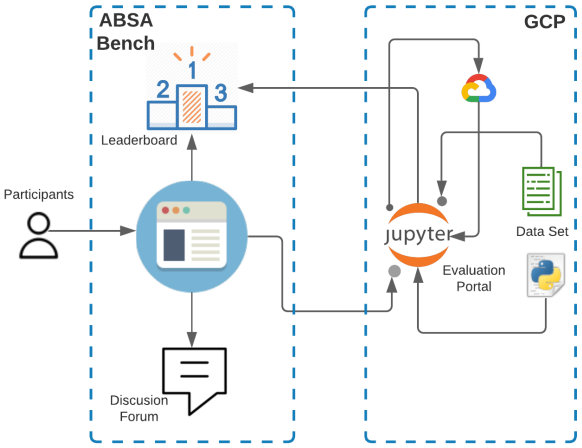


Figure 2: The Framework of ABSA-Bench

Once the trained model is submitted, it will get official scores on the test set. The platform also provides a documentation to help researchers understand how to use the platform. Please refer to Section 4.3.2 for more details.

### 4.3 The Web-based Platform

In order to enable the above-mentioned evaluations, we design and implement a Web-based benchmark platform that enables researchers to evaluate their ABSA models in a unified environment for fair comparison. The performances measured in Macro $F$-1 score is ranked in the leader board in the platform with a discussion board provided to exchange ideas among researchers. Specifically, the platform consists of three primary elements: *Leader board*, *Evaluation Portal*, and *Discussion forum*. Figure 2 shows these three elements in this platform.

#### 4.3.1 Leader Board

We maintain a leader board in ABSA-Bench based on the evaluations of some of the state-of-the art ABSA models so far. The performances of the models that are submitted by the authors will be added to the leader board and assigned a proper ranking position. For a fair comparison, the BERT based and Non-BERT based models have been ranked separately with two tabs in the leader board.

#### 4.3.2 Evaluation Portal

The computation power is supported by Google Cloud Platform which will serve the JupyterHub that is integrated with our platform. A pre-configured environment dedicated to ABSA will be created for participants. This environment will support complex computations and provide a task bundle which contains necessary dependencies for

the task and the evaluations. Users need to create an account and be authenticated to participate in the challenge. They can train and evaluate their model in their own work spaces leveraging the resources provided and managed by system administrators who can test the submitted prediction files and assess the submitted models under a unified standard.

#### 4.3.3 Discussion forum

A discussion forum is provided for participants once they create their account.

This will serve as a collaborative environment where researchers can post queries and collaborate. It will be especially helpful for new academics making an initial start in this field. This will save immense time in resolving concerns through a collaborative effort.

## 5 Performance Comparison

Discussion on the dataset including the motivation for choice, the implementation settings for the experiments and an objective comparison of the results have been presented in this section.

### 5.1 Data

We adopted SemEval14 Task 4 (Pontiki et al., 2014) as the benchmarking dataset. This is because it is the only widely accepted benchmark dataset for ABSA and has successfully fostered ABSA research since its release. Although later SemEval competitions also contain ABSA tasks, those datasets are derived from the SemEval14 version with small updates that deviate the evaluation purpose from ABSA. Therefore, we retain the original version intending to be more focused.

In SemEval14 ABSA task 4, there are two domain-specific subsets for laptops and restaurants reviews respectively, consisting of over 6,000 sentences with aspect-level human-authored labels for evaluation. Each single or multi-word aspect term is assigned one of the following polarities based on the sentiment that is expressed in the sentence towards it: positive, negative, neutral, and conflict. *Restaurants* includes annotations for coarse aspect categories, aspect terms, aspect term- specific polarities, and aspect category-specific polarities. *Laptop* includes annotations for aspect terms and their polarities. We removed the data with conflict sentiment polarity and the ones without aspect terms, obtaining 1,978 training samples and 600 test for *Restaurants* and 1,462 training samples and 411 test samples for *Laptop* respectively.

| Models | *Restaurants* | *Laptop* |
|---|---|---|
| CNN | 60.25 | 57.75 |
| LSTM | 65.51 | 55.35 |
| TD-LSTM | 68.98 | 61.87 |
| TC-LSTM | 66.72 | 61.11 |
| ATAE-LSTM | 63.72 | 58.47 |
| CABASC | **68.02** | **62.94** |
| IAN | 65.12 | 60.90 |
| RAM | 66.76 | 59.73 |
| MemNet | 61.09 | 58.01 |
| AEN-BERT | 73.76 | 76.31 |
| BERT-PT | 76.96 | 75.08 |
| BERT-SPC | 73.03 | 72.63 |
| LCF-BERT | **81.74** | **79.59** |

Table 1: Performances Comparison ($F$-1 in %) on the Unified Environment

### 5.2 Implementation Adjustment

We evaluated some of the state-of-the-art ABSA models as introduced in Section 3.1. To provide a unified computation environment, we made necessary adjustments and expect researchers to follow these adjustments and submit their models to ABSA-Bench for fair comparisons.

For Non-BERT-based models, GloVe[10] is adopted as the pre-trained word embedding. We have uniformly adjusted the dimension of the hidden state vectors as 300 and position embedding as 100. We initialised the weight matrices with the uniform distribution $U(-0.1, 0.1)$, and the biases were initialised to zero. We experimented with a couple of optimizers and finally selected Adam for all the models to maintain uniformity. We kept the learning rate as $2e-5$ and used $1e-5$ as the value of the $L_2$ regularisation parameter.

For BERT-based models, we used a pre-trained BERT[11] model to generate word vectors of sequences. All the models were implemented using Pytorch framework. Optimal parameters were selected during the training stage and the best performed models were selected for evaluation. We kept the default settings for other parameters as set in the original papers of each work.

### 5.3 Results

We report the evaluation results in this section, including prediction performance, run-time statistics and model sizes comparisons.

Table 1 reports the Macro $F1$ score in % of the examined models. We have compared BERT based models and Non-BERT based models separately as BERT based models have larger model sizes.

[10]https://nlp.stanford.edu/projects/glove/

[11]https://github.com/google-research/bert

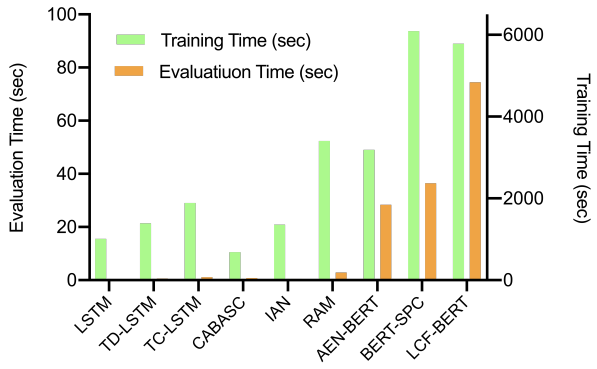


Figure 3: Model Run-Time Comparison

| Models | Params $10^6$ | Memory (MB) |
|---|---|---|
| CNN | 1.21 | 10.01 |
| LSTM | 7.23 | 35.61 |
| TD-LSTM | 1.44 | 12.41 |
| TC-LSTM | 2.16 | 14.11 |
| ATAE-LSTM | 2.53 | 16.61 |
| CABASC | 1.53 | 12.61 |
| IAN | 2.16 | 16.18 |
| RAM | 6.13 | 31.18 |
| MemNet | 0.36 | 7.8 2 |
| AEN-BERT | 112.93 | 451.84 |
| BERT-PT | 110 | 450.23 |
| BERT-SPC | 109.48 | 450.58 |
| LCF-BERT | 113.61 | 452.62 |

Table 2: Mode Size Comparison

BERT-based models achieved a much higher $F1$ score in comparison to Non-BERT based models as did for all the other NLP tasks. LCF-BERT model provided the best performance among BERT based models in our experiments. Among all the Non-BERT base models, CABASC has obtained the highest $F1$ score on both datasets. TC-LSTM outperforms basic LSTM model The results confirm that the context attention mechanism is more effective than the position attention mechanism. IAN outperforms ATAE-LSTM as it not only models the context representation, but also models the aspect representation by using attentions mechanism.

Figure 3 illustrates the comparisons of the model run-time i.e training and evaluation time. Table 2 present the comparisons of the model sizes in terms of the number of parameters and the size of the memory used during model training. From Figure 3 and Table 2, we observe the huge differences in the model sizes and execution times between BERT-based and Non-BERT based models. It is worth noting that for our experiments and also in the original papers, pre-trained BERT models have been used and therefore the model run time signifies time taken for fine-tuning and down-streaming the BERT model for particular task.

### 5.4 Evaluation Discussion

*Difference in the performances.* Compared to the values provided by the original papers, the performances of the examined models under our benchmarking environment ABSA-Bench show different macro $F1$ scores for all the models. It is easy to understand that the differences are as results of the different data pre-processing, implementation settings and evaluation environment. However, it is difficult to compare the models by just referring the papers. For example, the Macro $F1$ value for RAM is 70.51% for *Laptop* in (Li et al.) while the Macro $F1$ value for RAM is 71.35% for the same dataset in (Zeng et al., 2019). Given a new model with 71.00% Macro $F1$ on *Laptop*, we could not know whether it is better than RAM or not. This inconsistency motivates us to build an evaluation process on under a unified settings. Our platform aims to overcome these inconsistencies.

*Trade-off between the performances and the computational costs.* While BERT based models overall performed much better than Non-BERT based models, it is computationally more expensive. Even though pre-trained BERT models were used in the experiments, there was a significant increase in the computational cost which was mainly due to the huge difference in the parameter size. These models also limits research to industrial or big-scale research labs while researchers without the access to large-scale computation will be constrained with their experiments.

## 6 Conclusion and Future work

In this work, we design and implement an ABSA benchmarking evaluation process by providing two means of online evaluations and a Web-based platform. Leader board and discussion forums are enabled to rank the state-of-the-art ABSA research and share research ideas respectively. We examined some recent models and compared their actual differences under the unified platform ABSABench. This platform will help to understand the implementation of different deep learning models performing the task of ABSA. This understanding can then be utilised to improve the existing models. We intend to update our benchmarking platform with new tasks and datasets which will encourage quantitatively-informed research and learning.

## Acknowledgments

This project is sponsored by Google Academic Research Grants.

## References

Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. LinCE: A Centralized Benchmark for Linguistic Code-switching Evaluation. In *Proc. of the LREC 2020*, pages 1803–1813, Marseille, France.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent Attention Network on Memory for Aspect Sentiment Analysis. In *Proc. of the EMNLP 2017*, pages 452–461, Copenhagen, Denmark.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question Answering in Context. In *Proc. of the EMNLP 2018*, pages 2174–2184, Brussels, Belgium.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of the NAACL-HLT 2019*, pages 4171–4186, Minneapolis, MN, USA.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent Twitter Sentiment Classification. In *Proc. of the ACL HLT*, pages 151–160, Portland, Oregon, USA.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. NRC-Canada-2014: Detecting Aspects and Sentiment in Customer Reviews. In *Proc. of the SemEval 2014*, pages 437–442, Dublin, Ireland.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. Transformation networks for target-oriented sentiment classification. In *Proc. of the ACL 2018*.

Qiao Liu, Haibin Zhang, Yifu Zeng, Ziqi Huang, and Zufeng Wu. 2018. Content Attention Model for Aspect Based Sentiment Analysis. In *Proc. of the WWW 2018*, pages 1023–1032, Lyon, France.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *Proc. of the EMNLP 2015*, pages 1412–1421, Lisbon, Portugal.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive Attention Networks for Aspect-Level Sentiment Classification. In *Proc. of the IJCAI 2017*, pages 4068–4074, Melbourne, Australia.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The Natural Language Decathlon: Multitask Learning as Question Answering. *CoRR*, abs/1806.08730.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proc. of the SemEval 2014*, pages 27–35, Dublin, Ireland.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. In *Proc. of the ACL 2018*, pages 784–789, Melbourne, Australia.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100, 000+ Questions for Machine Comprehension of Text. In *Proc. of the EMNLP 2016*, pages 2383–2392, Austin, USA.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Targeted sentiment classification with attentional encoder network. In *Proc of the ICANN 2019*, pages 93–103.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective LSTMs for Target-Dependent Sentiment Classification. In *Proc. of the COLING 2016*, pages 3298–3307, Osaka, Japan.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect Level Sentiment Classification with Deep Memory Network. In *Proc. of the EMNLP 2016*, pages 214–224, Austin, Texas, USA.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Proc. of the IJCAI 2015*, pages 1347–1352, Buenos Aires, Argentina.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE:A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *Proc. of the ICLR 2019*, New Orleans, LA, USA.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for Aspect-level Sentiment Classification. In *Proc. of the EMNLP 2016*, pages 606–615, Austin, Texas.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis. In *Proc. of the NAACL-HLT 2019*, Minneapolis, MN, USA.

Wei Xue and Tao Li. 2018. Aspect Based Sentiment Analysis with Gated Convolutional Networks. In *Proc. of the ACL 2018*, pages 2514–2523, Melbourne, Australia.

Biqing Zeng, Heng Yang, Ruyang Xu, Wu Zhou, and Xuli Han. 2019. LCF: A Local Context Focus Mechanism for Aspect-Based Sentiment Classification. *Applied Sciences*, 9:3389.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A Benchmarking Platform for Text Generation Models. In *Proc. of the SIGIR 2018*, pages 1097–1100, Ann Arbor, MI, USA.