RobustQA

Stanford CS224N Default Project

Swee Kiat, Lim
Department of Computer Science
Stanford University
sweekiat@stanford.edu

Abstract

In recent years, question-answering (QA) models have vastly improved and achieved superhuman standards in several benchmarks. Yet, these same superhuman models often do not perform well on out-of-distribution (OOD) datasets or tasks. In contrast, humans appear to easily and quickly generalize to new unseen domains. In this project, we aim to train a QA model that is able to perform well across different datasets, especially on OOD datasets. Specifically, we experiment with the use of adversarial training applied to a pretrained DistilBERT model. The adversarial training takes the form of a critic model that tries to classify the origin domain of the QA embedding. In addition to the regular QA loss, the QA model has the additional objective of fooling the critic model. This encourages the QA model to learn a domain-agnostic embedding, which we hope to help with generalization and robustness to OOD datasets.

1 Key Information to include

- Mentor: NA
- External Collaborators: NA
- Sharing project: NA

2 Introduction

Question-answering (QA) models have achieved superhuman or close-to-human standards in several recent datasets and tasks such as SQuAD [1] and NewsQA [2], via works such as ALBERT by Lan et al. [3] and SpanBERT by Joshi et al. [4].

However, such models often do not perform well on out-of-distribution (OOD) datasets, as seen from previous works where authors evaluate trained models on OOD tasks and observe significant performance degradation [5, 6, 7, 8]. In contrast, humans appear to easily and quickly generalize to new unseen domains, such as the ability to quickly understand and internalize fantasy and science fiction settings.

In light of this problem, recent works have emerged proposing various alternatives to improve generalization of QA models. These include better design of QA tasks [6], debiasing of samples used during training [9], multitask learning [10, 11] and other works.

Likewise in this project, we aim to train a QA model that is able to generalize to OOD datasets. Specifically, we experiment with the use of adversarial training applied to a pretrained DistilBERT model. The adversarial training takes the form of a critic model that tries to classify the origin domain of the QA embedding. In addition to the regular QA loss, the QA model has the additional objective of fooling the critic model. This encourages the QA model to learn a domain-agnostic embedding, which we hope to help with generalization and robustness to OOD domains.

3 Related Work

Various works have highlighted the poor generalization of supposedly successful QA models [5, 6, 7, 8]. For example, Talmor and Berant [5], as well as Sen and Saffari [6], both observe that QA models tend to overfit to the datasets they were trained on and generalize poorly to new domains, especially in the zero-shot setting. In particular, Talmor and Berant also demonstrate that one way to alleviate this problem is simply to train on larger and more diverse datasets comprising multiple domains.

On the other hand, adversarial training has been gaining popularity as a way to improve performance of models in different settings. In order to avoid confusion, we will distinguish between two general classes of algorithms that have both been termed “adversarial training" but are significantly different.

The first class of “adversarial training" refers to algorithms that make use of adversarial samples. Adversarial samples refer to samples that are intentionally perturbed to trick a trained model into generating incorrect outputs. For example Goodfellow et al. demonstrated how image classifiers can be tricked into wrongly labeling images with high confidence by adding noise that is imperceptible to humans via the fast gradient sign method (FGSM) [12]. Previous works have demonstrated that adding these adversarial samples into the training data can help to improve the robustness of models against these adversarial samples [13]. Hence the term “adversarial training" here refers to training on adversarial samples.

The second class of “adversarial training" refers to algorithms that typically use two models that compete in a minmax fashion. Works in this class often cite Goodfellow et al.’s work on generative adversarial networks (GANs) [14]. In the vanilla GAN example, a generator model creates images to fool a discriminator model, while the generator tries to distinguish between real and fake images. Both models are updated with opposing objectives and hence the “adversarial" nature of the algorithm.

Our work follows the second class of “adversarial training". In our context, the QA model is similar to the generator model, but outputs embeddings rather than images. Similarly, the critic model here is trained to classify the embeddings into the correct domain, rather than differentiate “real" and “fake" embeddings. This adversarial scheme can help improve generalization of models on OOD domains by encouraging the embeddings of the QA model to be domain-agnostic and hence prevent the QA model from overfitting onto the training set.

There are similar works in this setting that also employ adversarial training for improving multi-domain performance [15, 16, 17]. Sato et al. had previously applied adversarial training [16] to solve for a multilingual parsing task [18]. In their work, Sato et al. demonstrated that adversarial training was able to improve the performance of a graph-based parser on the multilingual parsing task. Our work is most similar to that of Lee et al., where the authors also applied adversarial training to multi-domain QA tasks [17]. Relative to Lee et al., our work goes into greater detail regarding the effects of scaling the weight of the adversarial loss term (see Section 4.2). We also adopt DistilBERT instead of BERT and show that the effects of adversarial training appears to generalize well to DistilBERT as well.

4 Approach

4.1 DistilBERT Basline

We first adopt the DistilBERT model as a baseline [19], which is a distilled version of the original BERT model [20]. Sanh et al. demonstrated that DistilBERT is smaller and faster while retaining most of the performance of the original BERT when evaluated against the GLUE benchmark [21]. In turn, BERT is itself a variation of the original Tranformer architecture [22].

Specifically, we use the DistilBERT QA implementation from the `huggingface` library [23] and perform finetuning on the QA datasets on top of the pretrained DistilBERT model. See Section 5.1 for details on the datasets used.

During finetuning, we optimize the DistilBERT model to predict the start and end locations of the answer span, as per a regular QA task. Hence, the loss function for the finetuning is essentially the sum of the cross-entropy losses for the start and end location predictions:

$$L_{QA} = -\log \mathbf{p}_{start}(i) - \log \mathbf{p}_{end}(j)$$

where $\mathbf{p}_{start}(i)$ and $\mathbf{p}_{end}(j)$ are the predicted probabilities of i being the start location and j being the end location respectively.

4.2 DistilBERT + Critic

We then apply adversarial training to the pretrained DistilBERT model via the addition of a critic model.

The critic model comprises three feedforward layers. The first two layers are 1024 dimensions and followed by ReLU activation, while the last layer comprise the logits and corresponds to the number of domains in the training set - 3 in our case (see Section 5.1). The input to the critic model are the embeddings from DistilBERT just before the final output layer.

At every iteration, we update the critic model to optimize cross-entropy loss and to correct classify the correct domains given a batch of DistilBERT embeddings. We then update the DistilBERT model with the regular QA loss and an additional adversarial loss term. The adversarial loss term is essentially the negative of the cross-entropy loss multiplied by a scaling factor α:

$$L_{adv} = -\log \mathbf{p}_{critic}(d)$$
$$L_{total} = L_{QA} - \alpha L_{adv}$$

where $\mathbf{p}_{critic}(d)$ is the predicted probability of d being the source domain of the given embedding.

Hence, this builds on top of the previous baseline with the addition of the adversarial loss term. In our experiments, we vary the scaling factor α from 0 to 0.5 to better understand the effect of the adversarial loss.

5 Experiments

5.1 Data

In this work, we primarily use the datasets stipulated in the RobustQA task. For training, this comprise of 3 in-domain datasets (Natural Questions [24], NewsQA [2] and SQuAD [1]) and a small amount of training data from 3 out-of-domain datasets (DuoRC [25], RACE [26], RelationExtraction [27]). For validation and test, we use the out-of-domain datasets (DuoRC [25], RACE [26], RelationExtraction [27]).

5.2 Evaluation method

Our evaluation methods seek to measure the performance of our models on the QA task in a domain-agnostic manner that is comparable with current benchmarks. To that end, we will be comparing and evaluating the models based on the Exact Match (EM) and F1 scores.

We will be evaluating the models on three settings.

1. **In-domain validation** - this is similar to a regular QA setting
2. **OOD validation** - this is akin to a zero-shot QA setting where the model never sees any OOD samples during training or finetuning
3. **OOD validation after finetuning on OOD training** - this allows the model to learn from a small amount of OOD training set

5.3 Experimental details

In our experiments, we primarily varied the scaling factor α of the adversarial loss term, in order to better understand the effects of adversarial training. The values of α in our experiments varied across

Table 1: Performance on in-domain and OOD validation datasets with varying α.

α	**In-Domain EM**	**In-Domain F1**	**OOD EM**	**OOD F1**
0 (baseline)	54.71	70.79	31.94	48.13
0.01	54.80	70.71	31.41	48.11
0.05	**54.96**	**70.92**	30.10	46.78
0.1	54.88	70.61	32.20	47.50
0.5	54.38	70.23	**32.46**	**49.05**

Table 2: Performance on OOD validation datasets after finetuning on small OOD training set.

α	**OOD EM**	**OOD F1**
0 (baseline)	31.41	48.44
0.01	31.15	47.32
0.05	**32.98**	49.17
0.1	**32.98**	47.84
0.5	32.46	**49.18**

$[0, 0.01, 0.05, 0.1, 0.5]$. The setting at $\alpha = 0$ is essentially the baseline setting with no adversarial loss.

Aside from varying α, we kept all other parameters constant. The critic model is detailed in Section 4.2. In all experiments, we train with a batch size of 16 and a learning rate of 3e-5 with the AdamW optimizer [28]. We train for 6 epochs while evaluating on the in-domain validation set every 5000 iterations. We keep the model with the highest validation score on the in-domain validation set.

Finally, we further finetune the trained model on the much smaller OOD training set. We finetune on the OOD training set for 10 epochs and retain the model with the highest validation score on the OOD validation set.

We report results both before and after finetuning on the smaller OOD training set. The results prior to finetuning on the OOD training set can also be seen as zero-shot performance on the OOD task.

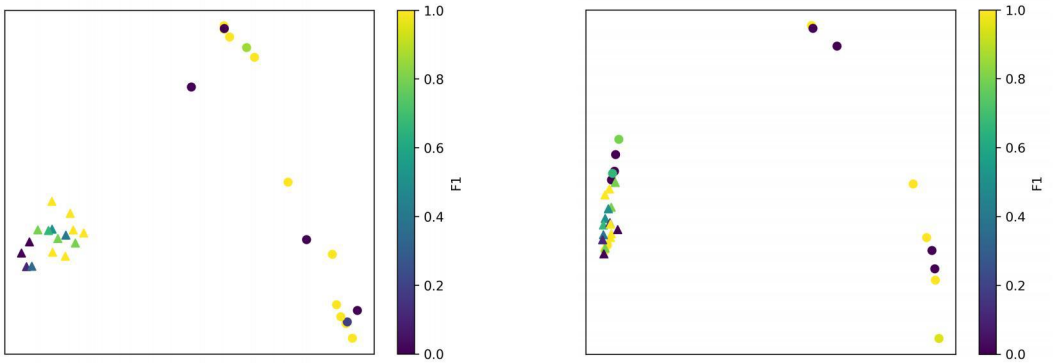
5.4 Results

Table 1 shows the performance of the models on in-domain and out-of-distribution (OOD) validation sets, where the models are trained with varying levels of the scaling factor α, prior to finetuning on the OOD training set. Table 2 shows the performance on OOD validation sets after finetuning on the OOD training set.

As expected, we observe that finetuning on the OOD training set helps with performance on the OOD task. Furthermore, of interest to this work, we see that finetuning on the OOD training set appears to have a larger effect on the models that were trained with adversarial loss. This suggests that training with adversarial loss does help with generalization to unseen domains and improves the ability of pretrained models to perform transfer learning with limited data.

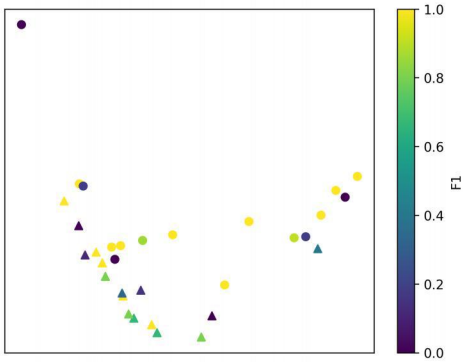
In addition, we also see that adversarial training does not harm the models' performance on the in-domain validation set. We can see in the first two columns of Table 1 that in-domain performance is preserved even with adversarial training. In some cases (e.g. $\alpha = 0.05$), adversarial training even helps improve performance on the in-domain validation set. Intuitively, adversarial training encourages the QA model to generate domain-agnostic representations. This may also serve to prevent overfitting on the training set and hence improve performance on the in-domain validation set.

Our final results on the test set leaderboard used the model trained with $\alpha = 0.05$ after finetuning on the OOD training set and achieved EM of 41.743 and F1 of 59.899.

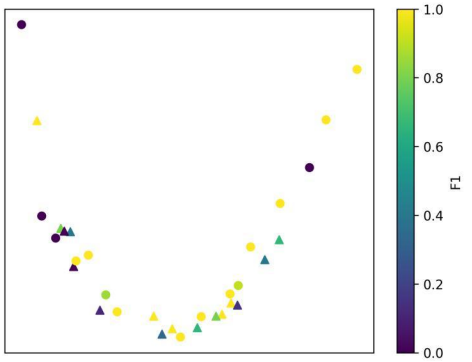


(a) Without adversarial training ($\alpha = 0$).

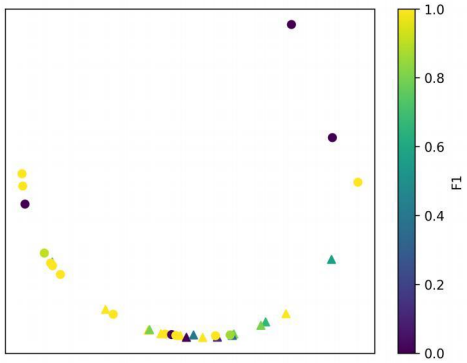
(b) With adversarial training ($\alpha = 0.01$).



(c) With adversarial training ($\alpha = 0.05$).



(d) With adversarial training ($\alpha = 0.1$).



(e) With adversarial training ($\alpha = 0.5$).

Figure 1: PCA visualization of DistilBERT embeddings of question-context pairs, color-coded by their F1 values. Circles represent in-distribution samples and triangles represent OOD samples. In-domain versus OOD clustering is clearly observed in the case of no adversarial training where $\alpha = 0$. Some clustering is also observed when α is small at $\alpha = 0.01$.

6 Analysis

Next, we perform analysis of the samples via PCA visualization of DistilBERT embeddings of question-context pairs. Figure 1 plots the embeddings color-coded by their F1 values, where circles represent in-distribution samples and triangles represent OOD samples.

Specifically, we randomly sample 16 question-context pairs each, from the NewsQA validation dataset for in-domain and RelationExtraction validation dataset for OOD. We then use PCA to project the DistilBERT embeddings of the question-context pairs onto a 2D space for visualization. We do

this for all the models with varying α from 0 to 0.5. In all cases, we use the models prior to finetuning on the OOD training set.

In Figure 1a, we see that without adversarial training, there is obvious clustering of the OOD samples (triangles) on the left and in-domain samples (circles) on the right. Furthermore, we see that the OOD samples closer to the in-domain samples also tend to have higher F1, as shown by the yellow triangles in the upper right of the OOD cluster. In contrast, the bottom left triangles in the OOD cluster are further away from the in-domain samples and also tend to have much lower F1. This implies that the model may have overfitted to the in-domain task and hence unable to generalize well to OOD samples that are too different from the in-domain samples.

On the other hand, Figures 1b to 1e shows that with adversarial training, the clustering gradually disappears with increasing α. Some clustering is still present at $\alpha = 0.01$ but both OOD samples and in-domain samples appear well-mixed as α increases past 0.01. This clearly demonstrates that the DistilBERT QA model is indeed able to learn domain-agnostic representations via the adversarial loss term and the critic model, even with scaling factor as small as $\alpha = 0.05$.

7 Conclusion

In this work, we focus on the use of adversarial training for improving generalization of QA models to OOD domains. By varying the scaling factor applied to the adversarial loss term, we show that adversarial training does improve generalization to OOD domains without degrading performance on the in-domain task.

Furthermore, our analysis in Section 6 shows that there is a trend where the model performs better on OOD samples that are closer to the in-domain samples (see Figure 1a). We also show qualitatively that the embeddings learned with adversarial training are more domain-agnostic, with no apparent clustering between in-domain versus OOD samples.

This work is primarily limited by the range of datasets and model architectures. A more comprehensive work may consider similar experiments on a larger variety of datasets and model architectures. Furthermore, future work may look at how adversarial training can complement other forms of generalization techniques to improve performance on OOD domains.

References

[1] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

[2] Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*, 2016.

[3] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

[4] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020.

[5] Alon Talmor and Jonathan Berant. MultiQA: An empirical investigation of generalization and transfer in reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4911–4921, Florence, Italy, July 2019. Association for Computational Linguistics.

[6] Priyanka Sen and Amir Saffari. What do models learn from question answering datasets? *arXiv preprint arXiv:2004.03490*, 2020.

[7] Mark Yatskar. A qualitative comparison of CoQA, SQuAD 2.0 and QuAC. 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 2318–2323, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[8] Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. Mrqa 2019 shared task: Evaluating generalization in reading comprehension. *arXiv preprint arXiv:1910.09753*, 2019.

[9] Mingzhu Wu, Nafise Sadat Moosavi, Andreas Rücklé, and Iryna Gurevych. Improving qa generalization by concurrent modeling of multiple biases. *arXiv preprint arXiv:2010.03338*, 2020.

[10] Dan Su, Yan Xu, Genta Indra Winata, Peng Xu, Hyeondey Kim, Zihan Liu, and Pascale Fung. Generalizing question answering system with pre-trained language model fine-tuning. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 203–211, 2019.

[11] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*, 2019.

[12] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[13] Xiaodong Liu, Hao Cheng, Pengcheng He, Weizhu Chen, Yu Wang, Hoifung Poon, and Jianfeng Gao. Adversarial training for large neural language models. *arXiv preprint arXiv:2004.08994*, 2020.

[14] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

[15] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

[16] Motoki Sato, Hitoshi Manabe, Hiroshi Noji, and Yuji Matsumoto. Adversarial training for cross-domain universal dependency parsing. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 71–79, 2017.

[17] Seanie Lee, Donggyu Kim, and Jangwon Park. Domain-agnostic question-answering with adversarial training. *arXiv preprint arXiv:1910.09342*, 2019.

[18] Daniel Zeman, Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinková, Jan Hajič jr., Jaroslava Hlaváčová, Václava Kettnerová, Zdeňka Urešová, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher D. Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria de Paiva, Kira Droganova, Héctor Martínez Alonso, Çağrı Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonça, Tatiana Lando, Rattima Nitisaroj, and Josie Li. CoNLL 2017 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada, August 2017. Association for Computational Linguistics.

[19] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[21] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[23] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.

[24] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

[25] Amrita Saha, Rahul Aralikatte, Mitesh M Khapra, and Karthik Sankaranarayanan. Duorc: Towards complex language understanding with paraphrased reading comprehension. *arXiv preprint arXiv:1804.07927*, 2018.

[26] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*, 2017.

[27] Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-shot relation extraction via reading comprehension. *arXiv preprint arXiv:1706.04115*, 2017.

[28] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.