CILDA: Contrastive Data Augmentation using Intermediate Layer Knowledge Distillation

Md Akmal Haidar[1]* **Mehdi Rezagholizadeh**[1] **Abbas Ghaddar**[1] **Khalil Bibi**[1]
Philippe Langlais[2] **Pascal Poupart**[3]
[1]Huawei Noah's Ark Lab
[2] RALI/DIRO, Université de Montréal, Canada
[3] David R. Cheriton School of Computer Science, University of Waterloo
{mehdi.rezagholizadeh,abbas.ghaddar}@huawei.com
felipe@iro.umontreal.ca, ppoupart@uwaterloo.ca

Abstract

Knowledge distillation (KD) is an efficient framework for compressing large-scale pre-trained language models. Recent years have seen a surge of research aiming to improve KD by leveraging **C**ontrastive Learning, **I**ntermediate **L**ayer Distillation, **D**ata Augmentation, and **A**dversarial Training. In this work, we propose a learning based data augmentation technique tailored for knowledge distillation, called CILDA. To the best of our knowledge, this is the first time that intermediate layer representations of the main task are used in improving the quality of augmented samples. More precisely, we introduce an augmentation technique for KD based on intermediate layer matching using contrastive loss to improve masked adversarial data augmentation. CILDA outperforms existing state-of-the-art KD approaches on the GLUE benchmark, as well as in an out-of-domain evaluation.

1 Introduction

The exponentially increasing size of pre-trained large language models (Devlin et al., 2019; Liu et al., 2020; Raffel et al., 2020; Brown et al., 2020) has been a persistent concern regarding the efficiency and scalability of Natural Language Understanding (NLU) in real world applications. Knowledge Distillation (KD) (Buciluǎ et al., 2006; Hinton et al., 2014) is a technique for transferring the knowledge from a large-scale model (called teacher) to a smaller one (called student), so that the latter model can be employed on edge device (Sanh et al., 2019a; Tang et al., 2019; Mukherjee and Awadallah, 2020; Li et al., 2021). This is done by minimizing the KL divergence between the teacher and student probabilistic outputs.

Numerous techniques have been exploited recently to increase the knowledge transfer beyond logits matching. For instance, it has been found beneficial to perform distillation on the internal components (parameters) of the teacher and student, which is known as Intermediate Layer Distillation (Sun et al., 2019, 2020b; Passban et al., 2021; Wang et al., 2020a,b; Fu et al., 2021; Wu et al., 2021).

Data Augmentation has also been successful for KD (Jiao et al., 2019; Shen et al., 2020; Qu et al., 2021), as researchers have found that the student has less opportunity to acquire useful information from the teacher when limited data are available for training (Kamalloo et al., 2021, 2022; Jafari et al., 2021a). Adversarial Training was also employed in KD (Zhu et al., 2019; Rashid et al., 2020, 2021; He et al., 2021) to improve the robustness and generalization, as the student may predict inconsistent outputs with slight distortion to the data distributions (Li et al., 2021). Recently, Contrastive Learning (Gutmann and Hyvärinen, 2010; Hjelm et al., 2018; Arora et al., 2019) has been exploited for improving knowledge transfer (Tian et al., 2019), and to optimize the intermediate layer mapping scheme (Sun et al., 2020a).

Each of the aforementioned techniques has proven effective in addressing a specific challenge in KD. Yet, we are not aware of a single method that takes advantage of all of them. In this paper, we propose CILDA, a KD method that incorporate **C**ontrasting Learning, **I**ntermediate **L**ayer Distillation, **D**ata Augmentation, and **A**dversarial Training. Distilling into a 6-layer BERT model, CILDA delivers new state-of-the-art results on the GLUE benchmark (Wang et al., 2018), as well as outperforming other KD methods in out-of-domain evaluations.

* Work done while at Huawei.

Proceedings of the 29th International Conference on Computational Linguistics, pages 4707–4713
October 12–17, 2022.

2 Related Work

Many studies (Jawahar et al., 2019; Tenney et al., 2019; Kovaleva et al., 2019) have noticed that important structural linguistic information are hidden in the intermediate layers of Transformer models (Vaswani et al., 2017). Recent KD methods propose to match teacher and student: intermediate layers representations (Jiao et al., 2019; Sun et al., 2019, 2020b; Wu et al., 2020), embedding matrix (Sanh et al., 2019a), and self-attention distributions (Wang et al., 2020a,b). Other variants of KD methods have been proposed such as Annealing-KD (Jafari et al., 2021b) and Pro-KD (Rezagholizadeh et al., 2021), two stage distillation methods where a smooth and gradual training of the student is controlled by a dynamic temperature factor, followed by a simple cross entropy loss for a few epochs.

Augmented adversarial examples (Miyato et al., 2016) are label-preserving transformations in the embedding space that are used to improve generalizability of models. FreeLB (Zhu et al., 2019) is an adversarial algorithm which creates virtual adversarial examples from word embeddings, and then performs the parameter updates on these adversarial embeddings. MATE-KD (Rashid et al., 2021) is a min-max adversarial data augmentation approach for KD, where an extra *generator* model is trained to generate adversarial text by maximizing the logit output margins between the teacher and the student.

Contrastive learning is a self-supervised representation learning method (Chen et al., 2020; Qu et al., 2021; van den Oord et al., 2018) which learns the feature representation of the samples by contrasting positive and negative samples. CODIR (Sun et al., 2020a) is a contrast-enhanced diversity promoting method between teacher and student intermediate representations of data samples from the same class. MATE-KD is the most related to our solution, with one notable difference: we believe our technique is the first to deploy intermediate layers distillation with the contrastive objective in the data augmentation process.

3 CILDA

In this section, we introduce CILDA, our contrastive approach for masked adversarial text augmentation for knowledge distillation using intermediate layer matching. Inspired by (Rashid et al., 2021), we deploy a generator (e.g. BERT) which will be trained to map masked inputs, $\tilde{X}$, to augmented samples, X'. The objective of this mapping is to perturb the inputs (in their vicinity) such that their corresponding output and intermediate layer representations of the teacher and student networks diverge to their maximum. Generating such maximum divergence augmented samples aims to fill the existing major gaps in the training data. We mask input tokens with a certain pre-defined probability, p. The architecture of our model is depicted in Figure 1. Our training is comprised of two alternating steps we describe hereafter.

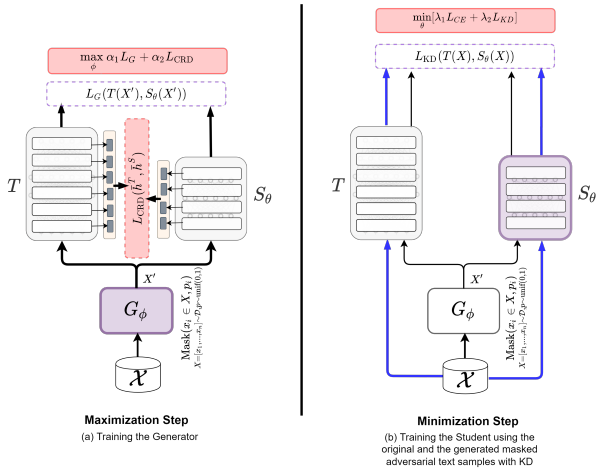

Figure 1: Illustration of maximization and minimization steps of CILDA.

Maximization Step: Generating Augmented Samples In the maximization step, the generator is trained in a way that the difference between the teacher and the student are maximized. As opposed to MATE-KD which only evaluates the divergence of the student and teacher networks based on their output, our technique takes intermediate layer matching into account as well. To the best of our knowledge this is the first time that the distance of intermediate layer representations are considered in the data-augmentation generation process. To be concise, MATE-KD only pays attention to the distance of samples in the output space, while our technique concerns the distance of samples in the input space as well. We hypothesize that to identify maximum divergence augmented samples, both input feature distances and output predictions are important. Our CILDA loss function to train

the generator can be described as:

$$L_{G_\phi} = \alpha_1 L_G + \alpha_2 L_{CRD}$$
$$L_G = KL\Big(\sigma(\frac{T(X')}{\tau_1}), \sigma(\frac{S_\theta(X')}{\tau_1})\Big) \quad (1)$$

where L_G is the KL-divergence loss between the teacher and the student logits, T and S_θ represent the teacher model and the student model with θ parameters respectively, σ is the softmax function and τ_1 is the temperature parameter that controls the softness of the output distributions, α_1 and α_2 are hyper-parameters. X' is the adversarial text output obtained by applying argmax to the generator output in the forward pass. Due to the non-differentiability issue of argmax in the backward pass, we use Gumbel-Softmax (Jang et al., 2016) at the output of the generator. More details can be found in (Rashid et al., 2021). L_{CRD} is the contrastive distillation loss that we introduced to the maximization step of MATE-KD. This contrastive loss is obtained by using the intermediate representation outputs of the teacher and the student models:

$$L_{CRD} = -\log \frac{exp(< \bar{h}_k^T, \bar{h}_k^{S_\theta} > /\tau_2)}{\sum_{j=0}^{K} exp(< \bar{h}_k^T, \bar{h}_j^{S_\theta} > /\tau_2)} \quad (2)$$

where τ_2 is the temperature parameter that controls the concentration level (Sun et al., 2020a). $\bar{h}_k^T$ and $\bar{h}_k^{S_\theta}$ are the intermediate layer representation of the teacher and student networks respectively, and $< ., . >$ is the cosine similarity between two feature vectors. k and j are indices of the samples of a mini-batch: k is the index of positive samples (i.e. the kth sample of the mini-batch is sent to both of the student and teacher networks to obtain their representations) and when $j \neq k$, we get negative samples (i.e. any other sample in the mini-batch excluding the kth sample) in a batch of K samples. The goal of this objective function is to map the student representations $\bar{h}_k^{S_\theta}$ of the positive sample k to $\bar{h}_k^T$, as well as the negative representations $\{\bar{h}_j^{S_\theta}\}_{j\neq k}^K$ far apart from $\bar{h}_k^T$.

For an arbitrary sample l in a mini-batch, the entire intermediate layer representations of the teacher and the student models (e.g. the $< CLS >$ representation of each layer of the networks) are concatenated to form $\hat{h}_l^T = [\bar{h}_{1,l}^T, \cdots, \bar{h}_{n,l}^T]$, $\hat{h}_l^{S_\theta} = [\bar{h}_{1,l}^{S_\theta}, \cdots, \bar{h}_{m,l}^{S_\theta}]$. Then these concatenated representations are further mapped into the same-size lower-dimensional spaces using linear projections $\bar{h}_l^T, \bar{h}_l^{S_\theta} \in R^u$ to calculate the distillation loss L_{CRD}. Here, n and m denote the number of intermediate layers of the teacher and the student networks respectively.

Minimization Step: Deploying Augmented Samples In the minimization step, the augmented adversarial samples produced by the generator and the training samples are used to minimize the difference between the teacher and the student. For this step, in the very general form, one can consider to match the student and teacher networks on their outputs and intermediate layer representations (e.g. using the contrastive loss) and the CE loss to match the output of the student with the labels:

$$L_{S_\theta} = \lambda_1 L_{CE} + \lambda_2 L_{KD} \quad (3)$$

where, L_{CE} describes the cross-entropy loss between the true label.

4 Experiments

4.1 Datasets and Evaluation

We experiment on 7 tasks from the GLUE benchmark (Wang et al., 2018): 2 single-sentence (CoLA and SST-2) and 5 sentence-pair (MRPC, RTE, QQP, QNLI, and MNLI) classification tasks. Following prior works, we report Pearson correlation on STS-B, Matthews correlation on CoLA, F1 score on MRPC, and use the accuracy otherwise. For out-of-domain evaluation, we report the performances on HANS (McCoy et al., 2019), SciTail (Khot et al., 2018), and IMDB using the models finetuned on MNLI, QQP, and SST-2 respectively.

4.2 Implementation Details

We use the 24-layer RoBERTa-large (Liu et al., 2020) and the 6-layer DistilRoBERTa (Sanh et al., 2019b) as the backbone for the teacher and the student models respectively. We perform hyperparameter tuning, and select best performing models using early stopping on dev sets. We use a linear transformation to map the intermediate representations into a 128-dimensional space and normalized them before computing the loss L_{CRD}. For each batch of data, we train the generator for n_G steps and the student model for $n_S = 100$ steps. We use $n_G = 20$ for CoLA, MRPC, RTE tasks and n_G=10 for the rest of the tasks. Following (Rashid et al., 2021), we set $p_{th} = 0.3$, $\alpha_1 = 1, \alpha_2 = 1, \tau_1 = 1.0, \tau_2 = 2.0$ for all of our

Model	CoLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	Avg.
					DEV				
Teacher	68.1	96.4	91.9	92.3	91.5	90.2	94.6	86.3	88.9
Vanilla-KD	60.9	92.5	90.2	89.0	91.6	84.1	91.3	71.1	83.8
Annealing-KD	61.7	93.1	90.6	89.0	91.5	85.3	92.5	73.6	84.7
MATE-KD	65.9	94.1	91.9	90.4	91.9	85.8	**94.6**	75.0	86.2
CILDA	**67.1**	**94.7**	**92.0**	**90.5**	**92.1**	**86.8**	92.9	**76.2**	**86.5**
					TEST				
Teacher	68.6	97.1	93.0	92.4	90.2	90.7	95.5	87.9	89.4
Vanilla-KD	54.3	93.1	86.0	85.7	89.5	83.6	90.8	74.1	82.1
Annealing-KD	54.0	93.6	86.0	86.8	89.7	84.4	90.8	73.7	82.4
MATE-KD	56.0	**94.9**	90.2	88.0	89.7	85.2	92.1	75.0	83.9
CILDA	**56.2**	**94.9**	**90.5**	**89.0**	**89.9**	**86.1**	**92.5**	**77.0**	**84.5**

Table 1: DEV and TEST performances on GLUE benchmark when RoBERTa_{24} and $\text{DistillRoberta}_{6}$ are used as backbone for the teacher and student variants respectively. Bold mark describes the best results.

experiments. We set λ_1 and λ_2 to 1/3 for the original training samples. For the augmented samples, we use $\lambda_2 = 2/9, \lambda_3 = 1/9$ for all tasks. The learning rate and the batch size are tuned from the set of {1e-5, 2e-5, 4e-6} and {8, 16, 32} respectively.

4.3 Results and Analysis

Table 1 shows the performances of the teacher, baselines, and our method on the GLUE dev and test sets. We compared CILDA to the Vanilla-KD (Hinton et al., 2014) baseline, and against 2 strong recently proposed methods [1]: Annealing-KD (Jafari et al., 2021b) and MATE-KD (Rashid et al., 2021). We observe that CILDA outperforms these models on all GLUE tasks, except on QNLI dev where MATE-KD performs better and SST-2 test where CILDA is on par with MATE-KD. On average over test sets, CILDA outperforms MATE-KD and Annealing-KD by a margin of 0.6% and 2.1% respectively.

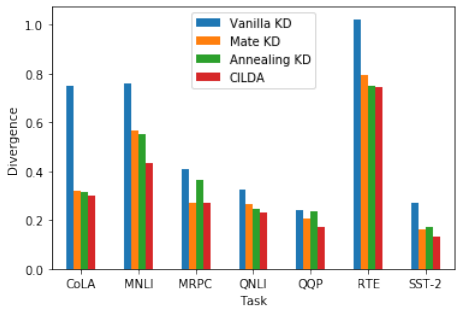

Figure 2: Divergence (lower is better) between the teacher and student logits on GLUE dev sets.

[1]We compare with these models because we have published results on GLUE leaderboard using the same teacher and student backbone models.

We investigate the logits generated by different methods to better understand why CILDA performs better. Figure 2 shows the divergence (lower is better) between the teacher and student logits on GLUE dev sets (except STS-B since it is a regression task) for 4 KD methods. Expectedly, Vanilla-KD (no enhancement) had the maximum divergence with teacher logits (which can be easily distinguished from other methods). We observe that CILDA mimic the teacher better than other methods on all tasks, which may partially explain the performance gains obtained by CILDA.

Model	HANS	PAWS	IMDB
Teacher	78.2	43.3	88.9
w/o KD	58.6	34.7	83.7
Vanilla-KD	58.9	36.5	84.0
Annealing-KD	61.2	35.8	84.6
MATE-KD	66.6	38.3	85.0
CILDA	**68.1**	**40.5**	**85.2**

Table 2: Out-of-domain performances of models trained on MNLI, QQP, SST-2 and evaluated on HANS, PAWS, and IMDB respectively.

Furthermore, we measure the robustness and generalization ability of the tested methods by evaluating them on out-of-domain test sets. Table 2 shows performances of models fine-tuned on MNLI, QQP, SST-2 and tested on HANS, PAWS, and IMDB respectively. CILDA significantly outperforms the second best method (MATE-KD) by 1.6% and 2.2% on HANS and PAWS respectively, and by a margin of 0.2% on IMDB.

5 Conclusion and Future Work

We proposed a min-max adversarial data augmentation framework for KD, which is powered by contrastive distillation loss for intermediate layer matching. Our algorithm maximizes the intermediate and logit representation margin between the teacher and the student models. In future works, we would like to investigate the distillation from super-large models such as Megatron (Shoeybi et al., 2019) and T5 (Raffel et al., 2020). Also, we would like to improve the generator output quality via distillation from generative models like GPT-2 (Radford et al., 2019).

Acknowledgments

We thank Mindspore[2] for the partial support of this work. We thank the anonymous reviewers for their insightful comments.

References

Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. 2019. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *https://arxiv.org/abs/1810.04805*.

Hao Fu, Shaojun Zhou an Qihong Yang, Junjie Tang an Guiquan Liu, Kaikui Liu, and Xiaolong Li. 2021. Lrc-bert: Latent-representation contrastive knowledge distillation for natural language understanding. In *AAAI*.

[2]A new deep learning computing framework https://www.mindspore.cn/

Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings.

Xuanli He, Islam Nassar, Jamie Kiros, Gholamreza Haffari, and Mohammad Norouzi. 2021. Generate, annotate, and learn: Generative models advance self-training and knowledge distillation. *arXiv preprint arXiv:2106.06168*.

Geoffrey Hinton, Oriol Vinyals, and Jeff. Dean. 2014. Distilling the knowledge in a neural network. *NIPS Workshop, https://arxiv.org/abs/1503.02531*.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2018. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*.

Aref Jafari, Mehdi Rezagholizadeh, and Ali Ghodsi. 2021a. Knowledge distillation by utilizing backward pass knowledge in neural networks. US Patent App. 17/359,463.

Aref Jafari, Mehdi Rezagholizadeh, Pranav Sharma, and Ali Ghodsi. 2021b. Annealing knowledge distillation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2493–2504.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. Tinybert: Distilling bert for natural language understanding. *arXiv preprint arXiv:1909.10351*.

Ehsan Kamalloo, Mehdi Rezagholizadeh, and Ali Ghodsi. 2022. When chosen wisely, more data is what you need: A universal sample-efficient strategy for data augmentation. *arXiv preprint arXiv:2203.09391*.

Ehsan Kamalloo, Mehdi Rezagholizadeh, Peyman Passban, and Ali Ghodsi. 2021. Not far away, not so close: Sample efficient nearest neighbour data augmentation via minimax. *arXiv preprint arXiv:2105.13608*.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *AAAI*.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of bert. *arXiv preprint arXiv:1908.08593*.

Tianda Li, Ahmad Rashid, Aref Jafari, Pranav Sharma, Ali Ghodsi, and Mehdi Rezagholizadeh. 2021. How to select one among all? an extensive empirical study towards the robustness of knowledge distillation in natural language understanding. *arXiv preprint arXiv:2109.05696*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Roberta: A robustly optimized fbertg pretraining approach. *arXiv preprint arXiv:1907.11692*.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*.

Subhabrata Mukherjee and Ahmed Hassan Awadallah. 2020. Xtremedistil: Multi-stage distillation for massive multilingual models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2221–2234.

Peyman Passban, Yimeng Wu, Mehdi Rezagholizadeh, and Qun Liu. 2021. Alp-kd: Attention-based layer projection for knowledge distillation. In *AAAI*.

Yanru Qu, Dinghan Shen, Yelong Shen, Sandra Sajeev, Jiawei Han, and Weizhu Chen. 2021. Coda: Contrast-enhanced and diversity promoting data augmentation for natural language understanding. In *ICLR*.

A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Ahmad Rashid, Vasileios Lioutas, Abbas Ghaddar, and Mehdi Rezagholizadeh. 2020. Towards zero-shot knowledge distillation for natural language processing. *arXiv preprint arXiv:2012.15495*.

Ahmad Rashid, Vasileios Lioutas, and Mehdi Rezagholizadeh. 2021. MATE-KD: Masked adversarial TExt, a companion to knowledge distillation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1062–1071, Online. Association for Computational Linguistics.

Mehdi Rezagholizadeh, Aref Jafari, Puneeth Salad, Pranav Sharma, Ali Saheb Pasand, and Ali Ghodsi. 2021. Pro-kd: Progressive distillation by following the footsteps of the teacher. *arXiv preprint arXiv:2110.08532*.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. 2019a. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019b. Distilroberta, a distilled version of roberta: smaller, faster, cheaper and lighter. https://huggingface.co/distilroberta-base.

Dinghan Shen, Mingzhi Zheng, Yelong Shen, Yanru Qu, and Weizhu Chen. 2020. A simple but tough-to-beat data augmentation approach for natural language understanding and generation. *arXiv preprint arXiv:2009.13818*.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. 2019. Patient knowledge distillation for bert model compression. *https://arxiv.org/abs/1908.09355*.

Siqi Sun, Zhe Gan, Yu Cheng, Yuwei Fang, Shuohang Wang, and Jingjing Liu. 2020a. Contrastive distillation on intermediate representations for language model compression. In *EMNLP*.

Zhiqing Sun, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. 2020b. Mobilebert: a compact task-agnostic bert for resource-limited devices. *arXiv preprint arXiv:2004.02984*.

Raphael Tang, Yao Lu, Linqing Liu, Lili Mou, Olga Vechtomova, and Jimmy Lin. 2019. Distilling task-specific knowledge from bert into simple neural networks. *arXiv preprint arXiv:1903.12136*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*.

Y. Tian, D. Krishnan, and P. Isola. 2019. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

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

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. 2020a. Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers. *arXiv preprint arXiv:2012.15828*.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020b. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *arXiv preprint arXiv:2002.10957*.

Yimeng Wu, Peyman Passban, Mehdi Rezagholizade, and Qun Liu. 2020. Why skip if you can combine: A simple knowledge distillation technique for intermediate layers. *arXiv preprint arXiv:2010.03034*.

Yimeng Wu, Mehdi Rezagholizadeh, Abbas Ghaddar, Md Akmal Haidar, and Ali Ghodsi. 2021. Universal-kd: Attention-based output-grounded intermediate layer knowledge distillation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7649–7661.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2019. Freelb: Enhanced adversarial training for natural language understanding. In *ICLR*.