Overview of the SustaiNLP 2020 Shared Task

Alex Wang
New York University
alexwang@nyu.edu

Thomas Wolf
HuggingFace
thomas@huggingface.co

Abstract

We describe the SustaiNLP 2020 shared task: efficient inference on the SuperGLUE benchmark (Wang et al., 2019). Participants are evaluated based on performance on the benchmark as well as energy consumed in making predictions on the test sets. We describe the task, its organization, and the submitted systems. Across the six submissions to the shared task, participants achieved efficiency gains of 20× over a standard BERT (Devlin et al., 2019) baseline, while losing less than an absolute point in performance.

1 Introduction

While ever-larger pretrained language models have led to impressive gains across a variety of natural language processing (NLP) tasks, there is growing concern about the environmental impact of training and deploying these models (Strubell et al., 2019; Schwartz et al., 2019). In response, there has been a growing body of research focusing on making these large models smaller and more efficient with minimal sacrifice to performance (Sanh et al., 2019; Michel et al., 2019, i.a.).

The SustaiNLP 2020 shared task focuses on the development of computationally and energy efficient NLP systems. The task uses the SuperGLUE benchmark (Wang et al., 2019), a standard benchmark for natural language understanding. Systems are evaluated on both the benchmark score as well as the energy consumed in evaluating the system on the benchmark. Participants are therefore incentivized to develop models that are energy efficient while maintaining the high performance of recent models. The shared task received six submissions that employed a large variety of optimizations to improve system efficiency. Overall, the submitted systems were on average 20× more efficient than a standard baseline using pretrained language models while nearly matching baseline performance.

2 Shared Task Description

2.1 Task

The shared task centers on the SuperGLUE benchmark, a suite of eight diverse NLU tasks designed to test a system's ability to perform a broad range of language understanding capabilities. The tasks vary substantially in task type, input size, and textual domain. We use seven of the eight SuperGLUE tasks, as the extremely small nature of the Winograd Schema Challenge (WSC) makes it challenging to obtain meaningful performance while improving the efficiency of the system. We briefly describe the seven tasks used here; see Wang et al. (2019) for an in-depth discussion of the tasks.

- Boolean Questions (BoolQ; Clark et al., 2019) is a question answering (QA) dataset where each example consists of a paragraph and a yes/no question about that paragraph. The test set consists of 3245 examples, and the evaluation metric is accuracy.

- CommitmentBank (CB; De Marneffe et al., 2019) is a natural language inference (NLI) task where each example consists of a short text containing an embedded clause. The task is to determine if the embedded clause is entailed or contradicted by the original text. The test set consists of 250 examples, and the evaluation metrics are accuracy and F1.

- Choice of Plausible Alternatives (COPA; Roemmele et al., 2011) is a causal reasoning dataset where each example consists of a premise sentence and the task is to determine a likely cause or effect of the premise from among two choices. The test set consists of 500 examples, and the evaluation metric is accuracy.

Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing, pages 174–178
Online, November 20, 2020.

- Multi-Sentence Reading Comprehension (MultiRC; Khashabi et al., 2018) is a QA dataset where each example consists of a paragraph and a variable number of multiple choice questions about the paragraph. Each questions can have one or more valid answers. The test set consists of 1800 examples, and the evaluation metrics are F1 over all answer choices as well as exact match of answer sets.

- Reading Comprehension with Commonsense Reasoning Dataset (ReCoRD; Zhang et al., 2018) is a QA dataset where each example consists of a news article and a Cloze question about the article whose answer choices are entities in the article. If an entity appears multiple times in the article, all mentions are considered correct. The test set consists of 10K examples, and the evaluation metrics are maximum token-level F1 (over all mentions) and exact match.

- Recognizing Textual Entailment (RTE; Dagan et al., 2006; Bar Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009) is a collection of NLI datasets where each example consists of a premise sentence and a hypothesis sentence. The task is to determine if the premise entails, contradicts, or is neutral to the hypothesis. The test set consists of 300 examples, and the evaluation metric is accuracy.

- Words in Context (WiC) is a word sense disambiguation task where each example consists of a pair of sentence that each contain the same marked word. The task is to determine if the word has the same sense in both sentences. The test set consists of 1400 examples, and the evaluation metric is accuracy.

To participate, each submission produces predictions on the test set of each task and is scored according to the task evaluation metrics. The overall task performance is determined by averaging performance metrics for each task. For tasks with multiple evaluation metrics, we first average within each task.

2.2 Efficiency

As the workshop focuses on developing computationally efficient systems, we additionally evaluate systems by how efficiently they produce predictions on the test set. We focus on measuring efficiency during inference rather than training, as, in the current paradigm, models are trained only a handful of (expensive) times but used for inference many more times. Additionally, measuring efficiency during training is complicated by the widespread reliance on pretrained model components.

Though there are many metrics for measuring efficiency, we follow the recommendation of Henderson et al. (2020) and measure efficiency by the power consumed throughout the course of inference. To do so, we use the `experiment-impact-tracker` library Henderson et al. (2020).

2.3 Organization

We consider two[1] tracks: one using GPUs and one restricted to CPU only. All systems were welcome to use any programming language or libraries, but were run on standardized hardware environments. For the GPU track, participants had four Nvidia V100s (32GB) available to them, but all participants chose to use only one GPU due to the cost of parallelization overhead. We run all submissions three times and report the mean task and efficiency scores.

3 Submissions

We provided participants with a simple baseline that follows the standard paradigm of finetuning a pretrained language model to each task. For pretrained models, we use BERT-base (Devlin et al., 2019) and RoBERTa-large (Liu et al., 2019), as provided by the HuggingFace Transformers library (Wolf et al., 2019).

There were six submissions to the shared task, four submissions to the GPU track and two submissions to the CPU track. All submissions were provided by Kim and Hassan (2020). We provide a brief description of the six submissions below; see Kim and Hassan (2020) for in-depth descriptions. Systems 1-* are submissions to the GPU track and systems 3-* are submissions to the CPU track.

- 1-1: This submission employs optimizations at all levels. The model is first trained using

[1]Originally, we considered three tracks: one CPU track and two GPU tracks separated by performance thresholds. However, we only received submissions to two of the three tracks

	system	total	BoolQ	CB	COPA	MultiRC	ReCoRD	RTE	WiC
GPU	BERT-base	328.781	7.035	1.334	1.299	20.380	290.650	5.350	0.734
	RoBERTa-large	752.935	16.020	2.734	4.014	43.278	667.607	13.126	6.156
	1-1	16.169	0.639	0.230	0.010	2.972	12.170	0.260	0.095
	1-2	15.248	0.594	0.023	0.016	2.524	11.632	0.337	0.122
	1-3	19.953	1.615	0.046	0.049	4.559	12.661	0.677	0.345
	1-4	20.477	1.641	0.050	0.060	5.356	12.348	0.653	0.369
CPU	BERT-base	1449.018	21.698	2.296	4.515	62.951	1324.910	22.182	10.466
	3-1	65.570	1.548	0.056	0.060	4.111	58.756	0.639	0.399
	3-2	92.797	1.911	0.102	0.166	6.830	82.750	0.259	0.778

Table 1: Energy consumption ($\times 1000$) in kWh for various systems.

system	avg	BoolQ	CB	COPA	MultiRC	ReCoRD	RTE	WiC
BERT-base	64.5	76.5	82.2/87.6	50.8	69.5/18.6	58.1/57.4	68.5	69.1
1-1	63.6	74.0	79.3/86.0	58.0	65.7/17.9	56.6/55.8	66.4	66.0
1-2	63.8	74.0	79.3/86.0	58.0	67.5/18.8	56.6/55.8	66.4	66.0
1-3, 3-1	63.6	73.7	79.3/86.0	58.0	65.8/18.1	56.6/55.8	66.9	65.9
1-4, 3-2	63.8	73.7	79.3/86.0	58.0	67.6/18.4	56.6/55.8	66.9	65.9

Table 2: Task performance for various systems. For BoolQ, COPA, RTE, and WiC, the evaluation metric is accuracy. For CB, the evaluation metrics are accuracy and F1. For MultiRC, the evaluation metrics are answer-level F1 and exact match. For ReCoRD, the evaluation metrics are token-level F1 and exact match. The overall task performance is an unweighted average of performance across tasks.

both task-specific and task-agnostic knowledge distillation (Hinton et al., 2015) from the pretrained and finetuned BERT model. They then reduce the model sizes via network pruning (Karnin, 1990) and further decrease the memory footprint by using 16-bit precision. Finally, they improve the runtime by fusing specific operations using onnxruntime and using a large evaluation batch size.

- 1-2: This submission is the same as 1-1 except they use a modified model for MultiRC.
- 1-3: This submission is a hybrid system that uses the GPU only for ReCoRD due to its much larger size and CPU for all other tasks. It uses the same optimizations as 1-1.
- 1-4: This submission is the same as 1-3 except it uses the modified MultiRC model.
- 3-1: This submission uses the same models as 1-3, but runs only on CPUs. It includes additional CPU-specific optimizations such as 8-bit quantization for some matrix multiplications and optimzed number of CPU processes per task.
- 3-2: This submission uses the same models as 1-4, but only uses CPUs. It uses the same optimizations as 3-1.

4 Results

Energy and task results are respectively presented in Tables 1 and Table 2.

We find that the submitted systems are able to substantially improve total energy consumption over the baseline systems, as much as $20\times$ in both the GPU and CPU settings, while trading off less than one point average task performance. The differences tend to be larger in the CPU setting than the GPU setting, likely because large, unoptimized pretrained language models were developed to be run on GPUs. The improvements of the submitted systems vary wildly between tasks, and do not scale linearly in the size of the test set. On CB and COPA, two of the smallest datasets, the improvements are as much as $50-100\times$ in the GPU setting. On WiC and BoolQ, the improvements are a more modest $10\times$. Similarly, the improvements do not seem to scale in the size of the inputs, as improvements on the paragraph-input tasks (BoolQ, MultiRC, and ReCoRD) are frequently matched and dwarfed by

improvements on the sentence-level tasks.

Among the systems, we find that the hybrid submissions (1-3, 1-4) consistently consume more power than the GPU-only counterparts (1-1, 1-2). All of the submissions that use a GPU (1-*) substantially outperform those that do not (3-*), which is in large part due to the large test set for ReCoRD. We observe fairly high variance between similar systems (1-1 and 1-2; 1-3 and 1-4; 3-1 and 3-2). In the worst case, systems 3-1 and 3-2 only differ by the MultiRC model, but the energy consumption varies significantly. We attribute this variance to runtime differences in the environment.

Task performances are consistently around 2 absolute points lower in the submitted systems than the baseline, except for COPA, where the submitted systems outperform the baseline. However, given the large efficiency improvements over the baseline, this tradeoff seems favorable.

5 Conclusion

We describe the results of the SustaiNLP 2020 Shared Task. The six submissions were able to substantially improve over the baseline systems, obtaining improvements $20\times$ in energy consumption while only losing a point in performance. To achieve these results, the submissions employed efficiency optimizations at numerous levels, including model architecture, storage, and runtime, which hints at the rich design space for efficient machine learning models.

Acknowledgments

We thank Peter Henderson for developing the `experiment-impact-tracker` library and for guidance on using the library.

References

Roy Bar Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second PASCAL recognising textual entailment challenge.

Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. 2009. The fifth PASCAL recognizing textual entailment challenge.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of NAACL-HLT 2019*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Machine learning challenges. evaluating predictive uncertainty, visual object classification, and recognising tectual entailment*, pages 177–190. Springer.

Marie-Catherine De Marneffe, Mandy Simons, and Judith Tonhauser. 2019. The CommitmentBank: Investigating projection in naturally occurring discourse. To appear in proceedings of Sinn und Bedeutung 23. Data can be found at https://github.com/mcdm/CommitmentBank/.

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 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, pages 1–9. Association for Computational Linguistics.

Peter Henderson, Jieru Hu, Joshua Romoff, Emma Brunskill, Dan Jurafsky, and Joelle Pineau. 2020. Towards the systematic reporting of the energy and carbon footprints of machine learning. *arXiv preprint 2002.05651*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint 1503.02531*.

Ehud D Karnin. 1990. A simple procedure for pruning back-propagation trained neural networks. *IEEE transactions on neural networks*, 1(2):239–242.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 252–262.

Young Jin Kim and Hany Hassan. 2020. Fastformers: Highly efficient transformer models for natural language understanding. In *First Workshop on Simple and Efficient Natural Language Processing*.

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 1907.11692*.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, pages 14014–14024.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI Spring Symposium Series*.

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

Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2019. Green ai. *arXiv preprint arXiv:1907.10597*.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, pages 3266–3280.

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. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Sheng Zhang, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, Kevin Duh, and Benjamin Van Durme. 2018. ReCoRD: Bridging the gap between human and machine commonsense reading comprehension. *arXiv preprint 1810.12885*.