High Performance Natural Language Processing

Gabriel Ilharco[†] **Cesar Ilharco**[‡] **Iulia Turc**[‡]
Tim Dettmers[†] **Felipe Ferreira**[‡] **Kenton Lee**[‡]
[†]Paul G. Allen School of Computer Science & Engineering, University of Washington
[‡]Google Research
{gamaga,dettmers}@cs.washington.edu
{ilharco,iuliaturc,felipeg,kentonl}@google.com

Abstract

Scale has played a central role in the rapid progress natural language processing has enjoyed in recent years. While benchmarks are dominated by ever larger models, efficient hardware use is critical for their widespread adoption and further progress in the field. In this cutting-edge tutorial, we will recapitulate the state-of-the-art in natural language processing with scale in perspective. After establishing these foundations, we will cover a wide range of techniques for improving efficiency, including knowledge distillation, quantization, pruning, more efficient architectures, along with case studies and practical implementation tricks.

1 Tutorial Proposal

Recent advances in natural language processing (Radford et al. (2018); Devlin et al. (2018); Liu et al. (2019); Brown et al. (2020), among many others) have substantially improved model capabilities. Notably, pre-trained checkpoints can be fine-tuned without substantial task specific modifications to create powerful models for a wide range of tasks (Wang et al., 2018, 2019). For many applications, production systems with models up to date with the state-of-the-art are meeting high quality bars for adoption across a wide variety of language tasks.

However, the ever larger computational requirements of such cutting-edge models—which quickly approximates the scale of a trillion parameters (Lepikhin et al., 2020)—imposes challenges to their widespread adoption and further progress in the field. This has driven increasing attention to methods that allow more efficient use of hardware, through techniques such as knowledge distillation (Hinton et al., 2015; Turc et al., 2019), quantization (Shen et al., 2020; Zafrir et al., 2019), pruning (Sanh et al., 2020), and architectural changes (Kitaev et al. (2020); Wang et al. (2020b); Katharopoulos et al. (2020); Zaheer et al. (2020), among others). Altogether, these techniques are promising avenues for more efficient natural language processing.

This tutorial starts with an introduction covering recent trends in NLP with scale in perspective, and covers foundational knowledge such as the transformer architecture (Vaswani et al., 2017) and the fine-tuning paradigm. We then move to core techniques for improving efficiency, including knowledge distillation, quantization and pruning, later covering recent work on architectural improvements, focusing on the move towards self-attention with linear complexity. Then, we dive into case studies by examining specific models such as Iandola et al. (2020) and Sun et al. (2020). Finally, we end with practical implementation considerations including model and data parallelism, gradient accumulation and floating point precision, ending the tutorial with closing notes and a questions and answers section. We outline the structure of this tutorial in Table 1.

1.1 Type of the tutorial

Cutting edge.

1.2 Reading list

Fundamentals: Bahdanau et al. (2014); Vaswani et al. (2017); Devlin et al. (2018); Brown et al. (2020); Lepikhin et al. (2020); Nakkiran et al. (2019).

Core techniques: Hinton et al. (2015); Turc et al. (2019); Jiao et al. (2019); Shen et al. (2020); Zafrir et al. (2019); Frankle and Carbin (2018); Brix et al. (2020); Sanh et al. (2020).

Efficient attention: Beltagy et al. (2020); Kitaev et al. (2020); Wang et al. (2020b); Stickland

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts, pages 24–27
Online, November 19 - 20, 2020. ©2020 Association for Computational Linguistics
https://doi.org/10.18653/v1/P17

Section	Subsection	Duration
Introduction	Overview of the field with scale into perspective	10 min
Fundamentals	Self-attention and the transformer architecture	25 min
Core techniques	Knowledge distillation	15 min
	Quantization	15 min
	Pruning	15 min
Efficient attention	Towards linear complexity in attention	30 min
Case studies	Efficient language models	20 min
	Retrieval	10 min
Scaling in practice	Practical considerations for scaling NLP models	35 min
Final considerations	Closing notes, Q&A	5 min
Total	-	180 min

Table 1: Structure of the tutorial with duration of each section.

and Murray (2019); Correia et al. (2019); Vyas et al. (2020); Katharopoulos et al. (2020); Zaheer et al. (2020).

Case studies: Botha et al. (2017); So et al. (2019); Sun et al. (2020); Yan et al. (2020); Wang et al. (2020a); Iandola et al. (2020); Mehta et al. (2020); Reimers and Gurevych (2019); Khandelwal et al. (2019); Guu et al. (2020).

Scaling in practice : Micikevicius et al. (2017); Krizhevsky (2014); Sohoni et al. (2019); Kaplan et al. (2020); Lepikhin et al. (2020)

1.3 Authors

Gabriel Ilharco is a PhD candidate at the University of Washington, where he is advised by Ali Farhadi and Hannaneh Hajishirzi. Previously, he worked at Google Research. His research interests lie at the intersection of Natural Language Processing and Computer Vision. His previous experience in teaching includes the tutorial *Deep Learning for Natural Language Processing with Tensorflow*, at KDD 2019. http://gabrielilharco.com/

Cesar Ilharco is a Research Engineer at Google, developing ML models for News Intelligence & Realtime Event Understanding, where performance is important for efficient serving at large scale. He was a guest lecturer and industry partner at Harvard University (ML for knowledge reconciliation), and co-organized the tutorials *Deep Learning for Natural Language Processing with Tensorflow* (KDD 2019) and *Neural Structured Learning: Training neural networks with structured signals* (KDD 2020).

Iulia Turc is a Software Engineer at Google Research, currently working on transfer learning. Her past experience at Google includes federated learning and applied machine learning for various products. Previously, Iulia completed her master's degree at the University of Oxford where she focused on machine translation. http://www.iuliaturc.com.

Tim Dettmers is a PhD student at the University of Washington where he is advised by Luke Zettlemoyer. He also works as a visiting researcher at Facebook AI Research, Seattle. His main research interests are large scale NLP models and efficient deep learning. https://timdettmers.com/about

Felipe Tiengo Ferreira is a Senior Staff Software Engineer leading News Intelligence and Realtime Event Understanding, an applied research team across Mountain View, NYC, Paris, Vienna and Zurich. Felipe has an expertise in making complex systems—including NLP components—work in real-time at massive scale across different product areas at Google. https://research.google/people/FelipeGoldstein/

Kenton Lee is a Research Scientist at Google. His research spans several areas in NLP, including structured prediction, question answering, and transfer learning. Before joining Google Research, Kenton completed a PhD at the University of Washington while working with Luke Zettlemoyer. https://kentonl.com.

1.4 Prerequisites

- Math: Basic understanding of probability theory and linear algebra;
- Machine Learning: Basic familiarity with embeddings and sequence-to-sequence models. Familiarity with self-attention, transformers, and large-scale pretraining is desirable;

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jan A. Botha, Emily Pitler, Ji Ma, Anton Bakalov, Alex Salcianu, David Weiss, Ryan McDonald, and Slav Petrov. 2017. Natural language processing with small feed-forward networks. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.

Christopher Brix, Parnia Bahar, and Hermann Ney. 2020. Successfully applying the stabilized lottery ticket hypothesis to the transformer architecture. *arXiv preprint arXiv:2005.03454*.

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*.

Gonçalo M. Correia, Vlad Niculae, and André F. T. Martins. 2019. Adaptively sparse transformers. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

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

Jonathan Frankle and Michael Carbin. 2018. The lottery ticket hypothesis: Finding sparse, trainable neural networks.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training.

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

Forrest N Iandola, Albert E Shaw, Ravi Krishna, and Kurt W Keutzer. 2020. Squeezebert: What can computer vision teach nlp about efficient neural networks? *arXiv preprint arXiv:2006.11316*.

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.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. *arXiv preprint arXiv:2006.16236*.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2019. Generalization through memorization: Nearest neighbor language models.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

Alex Krizhevsky. 2014. One weird trick for parallelizing convolutional neural networks. *arXiv preprint arXiv:1404.5997*.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2020. Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*.

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. *CoRR*, abs/1907.11692.

Sachin Mehta, Marjan Ghazvininejad, Srinivasan Iyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2020. Delight: Very deep and light-weight transformer. *arXiv preprint arXiv:2008.00623*.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2017. Mixed precision training. *arXiv preprint arXiv:1710.03740*.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. 2019. Deep double descent: Where bigger models and more data hurt. *arXiv preprint arXiv:1912.02292*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

Victor Sanh, Thomas Wolf, and Alexander M Rush. 2020. Movement pruning: Adaptive sparsity by fine-tuning. *arXiv preprint arXiv:2005.07683*.

Sheng Shen, Zhen Dong, Jiayu Ye, Linjian Ma, Zhewei Yao, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. 2020. Q-bert: Hessian based ultra low precision quantization of bert. In *AAAI*, pages 8815–8821.

David R So, Chen Liang, and Quoc V Le. 2019. The evolved transformer. *arXiv preprint arXiv:1901.11117*.

Nimit Sharad Sohoni, Christopher Richard Aberger, Megan Leszczynski, Jian Zhang, and Christopher Ré. 2019. Low-memory neural network training: A technical report. *arXiv preprint arXiv:1904.10631*.

Asa Cooper Stickland and Iain Murray. 2019. Bert and pals: Projected attention layers for efficient adaptation in multi-task learning.

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

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: The impact of student initialization on knowledge distillation. *arXiv preprint arXiv:1908.08962*.

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.

Apoorv Vyas, Angelos Katharopoulos, and François Fleuret. 2020. Fast transformers with clustered attention.

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.

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*.

Hanrui Wang, Zhanghao Wu, Zhijian Liu, Han Cai, Ligeng Zhu, Chuang Gan, and Song Han. 2020a. Hat: Hardware-aware transformers for efficient natural language processing. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. 2020b. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*.

Zhongxia Yan, Hanrui Wang, Demi Guo, and Song Han. 2020. Micronet for efficient language modeling. *arXiv preprint arXiv:2005.07877*.

Ofir Zafrir, Guy Boudoukh, Peter Izsak, and Moshe Wasserblat. 2019. Q8bert: Quantized 8bit bert. *arXiv preprint arXiv:1910.06188*.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *arXiv preprint arXiv:2007.14062*.