Transfer Learning in NLP
NLPL Winter School

Thomas Wolf - HuggingFace Inc.

Overview

- ❏ Session 1: Transfer Learning - Pretraining and representations
- ❏ Session 2: Transfer Learning - Adaptation and downstream tasks
- ❏ **Session 3: Transfer Learning - Limitations, open-questions, future directions**

Sebastian Ruder

Matthew Peters

Swabha Swayamdipta

Many slides are adapted from **a Tutorial on Transfer Learning in NLP** I gave at NAACL 2019 with my amazing collaborators 👉

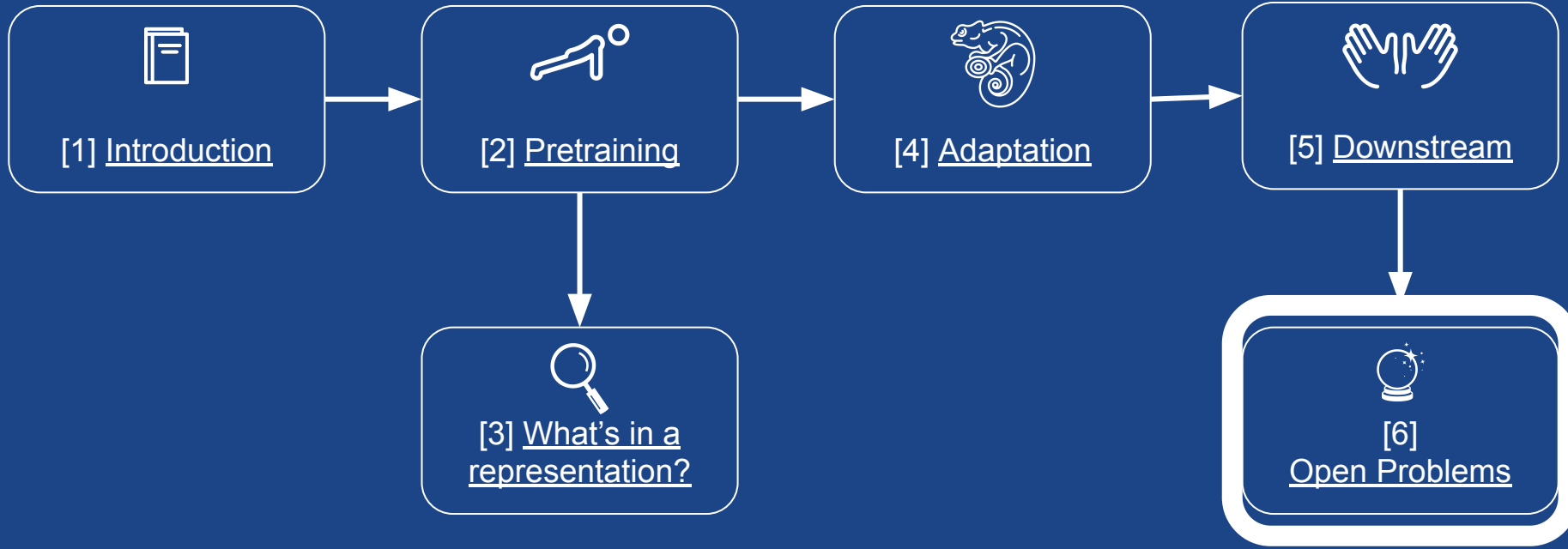
Transfer Learning in NLP
NLPL Winter School
Session 3

Transfer Learning in NLP

Follow along with the tutorial:

- Colab: https://tinyurl.com/NAACLTransferColab
- Code: https://tinyurl.com/NAACLTransferCode

Agenda


[1] Introduction
[2] Pretraining
[4] Adaptation
[5] Downstream
[3] What's in a representation?
[6] Open Problems

6. Open problems and future directions

Image credit: Yazmin Alanis

6. Open problems and future directions

A. Computation and model size
B. Lack of robustness
C. Reporting/evaluation issues
D. More data or better models?
E. In-domain generalization versus out-of-domain generalization
F. The limits of NLU and the rise of NLG
G. The question of inductive bias
H. The question of common-sense
I. Continual learning and meta-learning

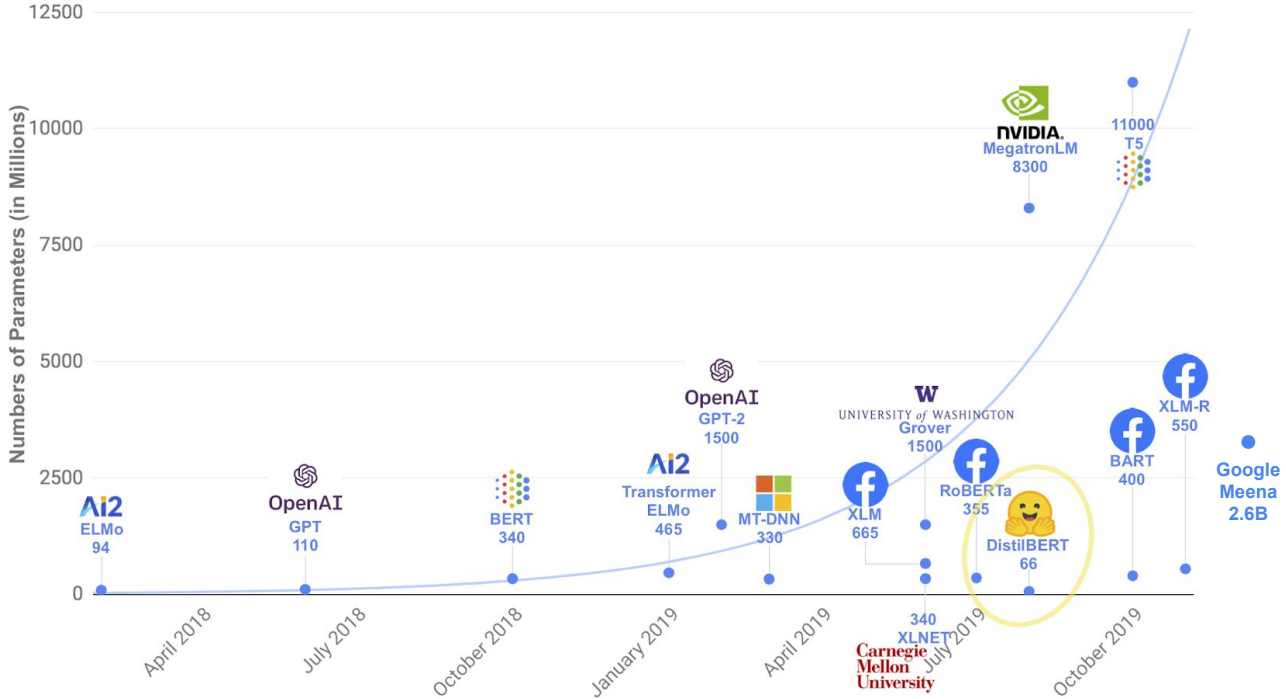
Image credit: Yazmin Alanis

Computation and model size

- Recent trends
  - Going big on model sizes
- Issues
  - Narrowing the research competition
  - Environmental costs
  - Is bigger-is-better a scientific research program?
- Going the other way
  - Models are over-parametrized
  - SustaiNLP competition
- Techniques
  - Distillation
  - Pruning
  - Quantization

Computation and model size

- Recent trends
  - as become the norm for SOTAGoing big on model sizes - over 1 billion parameters as become



Computation and model size

Why is this a problem?Why is this a problem?

- Narrowing the research competition field
  - what is the place of academia in today's NLP?
    fine-tuning? analysis and BERTology? critics?
- Environmental costs

"Energy and Policy Considerations for Deep Learning in NLP" - Strubell, Ganesh, McCallum - ACL 2019

Consumption	CO_2e (lbs)
Air travel, 1 passenger, NY↔SF	1984
Human life, avg, 1 year	11,023
American life, avg, 1 year	36,156
Car, avg incl. fuel, 1 lifetime	126,000
Training one model	
SOTA NLP model (tagging)	13
w/ tuning & experimentation	33,486
Transformer (large)	121
w/ neural architecture search	394,863

- Is bigger-is-better a scientific research program?

Rank	Name	Model	URL	Score
1	T5 Team - Google	T5		90.3
2	ERNIE Team - Baidu	ERNIE		90.0
3	Microsoft D365 AI & MSR AI & GATECH	MT-DNN-SMART		89.9
4	王玮	ALICE v2 large ensemble (Alibaba DAMO NLP)		89.7
5	Microsoft D365 AI & UMD	FreeLB-RoBERTa (ensemble)		88.4
6	Junjie Yang	HIRE-RoBERTa		88.3
7	Facebook AI	RoBERTa		88.1
8	Microsoft D365 AI & MSR AI	MT-DNN-ensemble		87.6

Computation and model size

Going the other way – smaller models

- Neural net are over parametrized

Optimal Brain Damage

Yann Le Cun, John S. Denker and Sara A. Solla
AT&T Bell Laboratories, Holmdel, N. J. 07733

LeCun, Y., Denker, J.S., & Solla, S.A. (1989). Optimal Brain Damage. *NIPS*.

Yu, Haonan et al. "Playing the lottery with rewards and multiple languages: lottery tickets in RL and NLP." *ArXiv* abs/1906.02768 (2019)

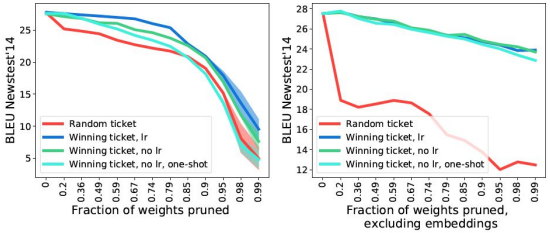


Figure 2: Winning ticket initialization performance for Transformer Base models trained on machine translation.

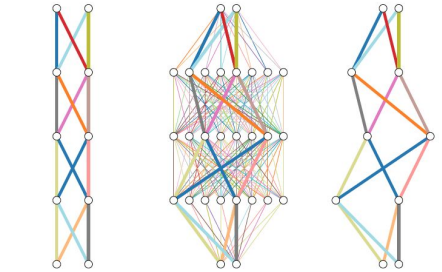


Figure 1. If a neural network with random weights (center) is sufficiently overparameterized, it will contain a subnetwork (right) that perform as well as a trained neural network (left) with the same number of parameters.

Ramanujan, Vivek et al. "What's Hidden in a Randomly Weighted Neural Network?" *ArXiv* abs/1911.13299 (2019): n. pag.

- Training sparse models for scratch – the GPU issue
  - Trading off speed/memory/flexibility
  - CPU/IPU?

GRAPHCORE

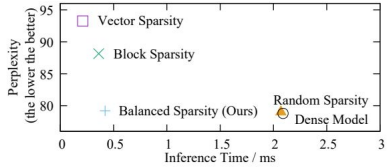
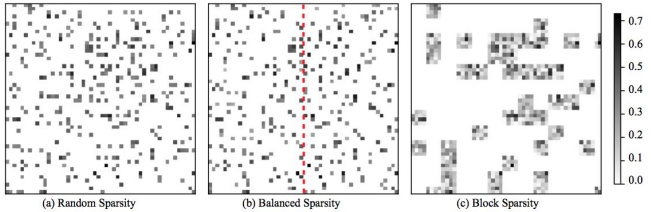


Figure 1: Perplexity and Inference Time trade-off of different sparsity patterns on the PTB dataset (Marcus et al. 1999). All the methods prune the same pre-trained LSTM models with single 1500-hidden-units cell to reach 90% sparsity.



Yao, Z., Cao, S., Xiao, W., Zhang, C., & Nie, L. (2018). Balanced Sparsity for Efficient DNN Inference on GPU. *AAAI*.

Computation and model size

Promoting smaller models

- Lack of incentive
  - Reviewing overload => focus on SOTA only
- SustaiNLP 2020 co-located with EMNLP 2020
  - First Workshop on Simple and Efficient Natural Language Processing
  - @sustainlp2020 – https://sites.google.com/view/sustainlp2020
- Shared task to stimulate the development of more efficient models
  - **Based on**: GLUE/SuperGLUE
  - **Goal**: optimal trade-off between performance and efficiency
  - **Evaluation**: ranking models according to efficiency under model performance constraints
  - Focus on **inference**
    - training efficiency difficult to fairly evaluate
    - training cost make headlines but… cumulative lifetime environmental cost of large-scale production models is mostly constituted by inference computational cost

Computation and model size

Reducing the size of a pretrained model

Three main **techniques** currently investigated:

- ❏ Distillation
- ❏ Pruning
- ❏ Quantization

Computation and model size

Distillation

- The best of both worlds (large models and small models)
  - reduce inference cost
  - capitalize on the inductive biases learned by a large model.
- **DistilBert**: 95% of Bert performances in a model 40% smaller and 60% faster

```
Input: ['[CLS]', 'i', 'think', 'this', 'is', 'the', 'beginning', 'of', 'a', 'beautiful', '[MASK]', '.', '[SEP]']
Rank 0  - Token: day             - Prob: 0.21348
Rank 1  - Token: life            - Prob: 0.18380
Rank 2  - Token: future          - Prob: 0.06267
Rank 3  - Token: story           - Prob: 0.05854
Rank 4  - Token: world           - Prob: 0.04935
Rank 5  - Token: era             - Prob: 0.04555
Rank 6  - Token: time            - Prob: 0.03210
Rank 7  - Token: year            - Prob: 0.01722
Rank 8  - Token: history         - Prob: 0.01663
Rank 9  - Token: summer          - Prob: 0.01335
Rank 10 - Token: adventure       - Prob: 0.01233
Rank 11 - Token: dream           - Prob: 0.01209
Rank 12 - Token: moment          - Prob: 0.01129
Rank 13 - Token: night           - Prob: 0.01084
Rank 14 - Token: beginning       - Prob: 0.00937
Rank 15 - Token: season          - Prob: 0.00664
Rank 16 - Token: journey         - Prob: 0.00621
Rank 17 - Token: period          - Prob: 0.00553
Rank 18 - Token: relationship    - Prob: 0.00517
Rank 19 - Token: thing           - Prob: 0.00508
```

$$L = -\sum_i t_i * log(s_i)$$

With **t** the logits from the teacher and **s** the logits of the student

To further expose the mass of the distribution over the classes, Hinton et al. introduce a **softmax-temperature**:

$$p_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)}$$

T is the temperature parameter.

Computation and model size

Distillation

- A lot of fresh work in late 2019
  Tsai et al., Turc et al., Tang et al.
- Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., & Liu, Q. (2019). TinyBERT: Distilling BERT for Natural Language Understanding. ArXiv, abs/1909.10351

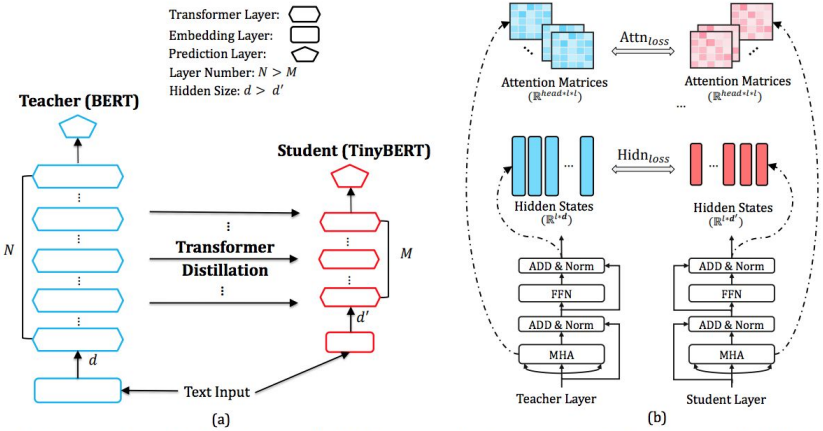


Figure 1: An overview of Transformer distillation: (a) the framework of Transformer distillation, (b) the details of Transformer-layer distillation consisting of Attn_{loss}(attention based distillation) and Hidn_{loss}(hidden states based distillation).

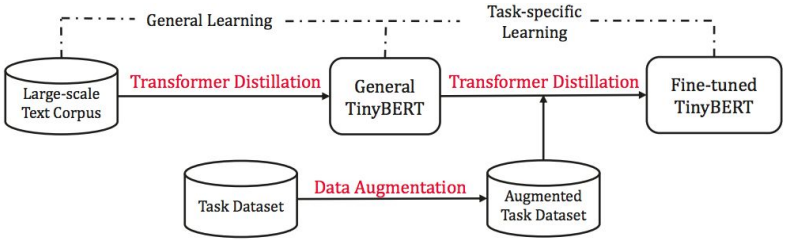
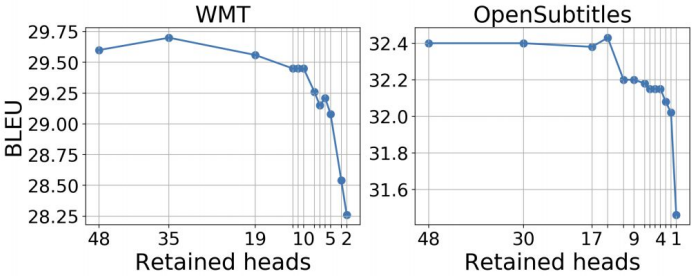


Figure 2: The illustration of TinyBERT learning

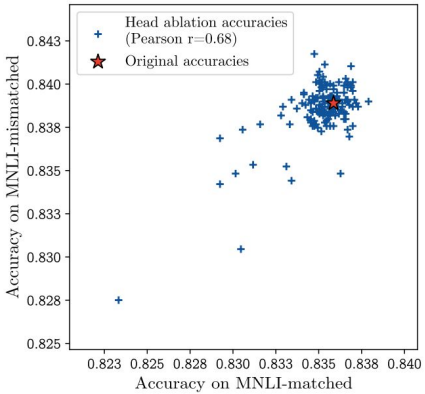
Computation and model size

Head pruning

- Elena Voita et al., “Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned,” *ArXiv:1905.09418 [Cs]*, May 22, 2019, http://arxiv.org/abs/1905.09418

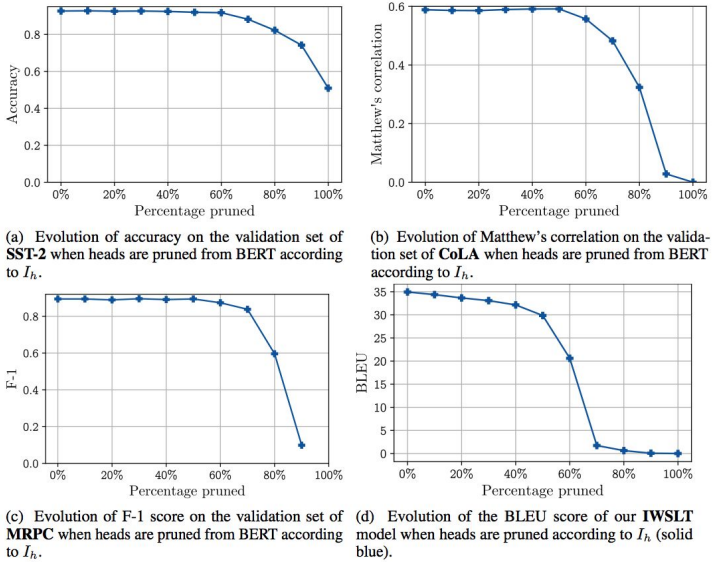


- Paul Michel, Omer Levy, and Graham Neubig, “Are Sixteen Heads Really Better than One?,” *ArXiv:1905.10650 [Cs]*, November 4, 2019, http://arxiv.org/abs/1905.10650.



Head Importance Score for Pruning

$$I_h = \mathbb{E}_{x \sim X} \left| \frac{\partial \mathcal{L}(x)}{\partial \xi_h} \right|$$



(a) Evolution of accuracy on the validation set of **SST-2** when heads are pruned from BERT according to I_h.

(b) Evolution of Matthew's correlation on the validation set of **CoLA** when heads are pruned from BERT according to I_h.

(c) Evolution of F-1 score on the validation set of **MRPC** when heads are pruned from BERT according to I_h.

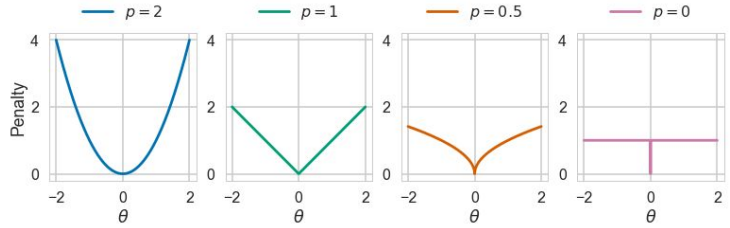
(d) Evolution of the BLEU score of our **IWSLT** model when heads are pruned according to I_h (solid blue).

Computation and model size

Weights pruning

- **ASAPP**: Ziheng Wang, Jeremy Wohlwend, and Tao Lei, “Structured Pruning of Large Language Models,” *ArXiv:1910.04732 [Cs, Stat]*, October 10, 2019, http://arxiv.org/abs/1910.04732
  - Low-rank matrix factorization + differential L0 pruning using a Hard Concrete distribution
  - RoBERTa on GLUE (99% performances)

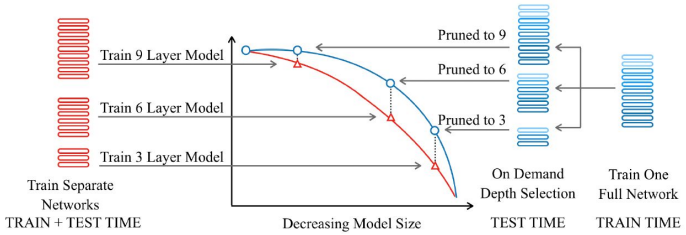
Parameters		SST2	MRPC	STS-B	QNLI	Average
125M	(100%)	92.43	90.9	90.22	89.77	**90.83**
80M	(65%)	92.09	88.61	88.18	89.05	**89.48**

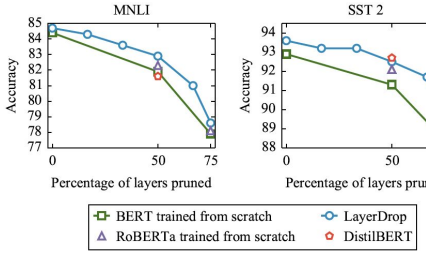


Christos Louizos, Max Welling, and Diederik P. Kingma, “Learning Sparse Neural Networks through L0 Regularization,” *ArXiv:1712.01312 [Cs, Stat]*, December 4, 2017, http://arxiv.org/abs/1712.01312.

Layer pruning

- **Facebook**: Angela Fan, Edouard Grave, and Armand Joulin, “Reducing Transformer Depth on Demand with Structured Dropout,” *ArXiv:1909.11556 [Cs, Stat]*, September 25, 2019, http://arxiv.org/abs/1909.11556.





	MNLI	SST2
6 Layers (50% Pruned)		
RoBERTa	82.3	92.1
+ LayerDrop	82.9	92.5
+ more data	**84.1**	**93.2**
3 Layers (75% Pruned)		
RoBERTa	78.1	90.3
+ LayerDrop	78.6	90.5
+ more data	**82.2**	**92.0**

Computation and model size

Quantization

- Quantized Tensors
  - From FP32 to INT8

$$Q(x, \text{scale}, \text{zero\_point}) = \text{round}\left(\frac{x}{\text{scale}} + \text{zero\_point}\right)$$

- Dynamic quantization on Bert
  - Applied on torch.nn.Linear — 438 MB FP32 => 181 MB INT8
  - (experimental) Dynamic Quantization on BERT
  - 0.6% F1 score accuracy after applying post-training dynamic quantization on fine-tuned BERT on the MRPC task

Prec	F1 score	Model Size	1 thread	4 threads
FP32	0.9019	438 MB	160 sec	85 sec
INT8	0.8953	181 MB	90 sec	46 sec

- Q8BERT (Intel), a Quantized 8bit Version of BERT-Base
  - https://www.intel.ai/q8bert/
  - Ex: MRPC F1 0.8788 with post-training dynamic quantization and 0.8956 with quantization-aware training.
  - Symmetric quantization: *Quantize(x, scale, bits) = Clip(Round(x * scale), - ($2^{bits - 1}$ - 1), $2^{bits - 1}$ - 1)*

Lack of robustness

- High variability - easy to fall in local minima
  - Bert on STILS: variability
  - Hyper parameter search for fine-tuning
- Solutions
  - Better regularization? (Mix-out)
  - Ensembles (distilled if necessary cf. Microsoft’s MT-DNN))

Lack of robustness

- High variability - easy to fall in local minima
  - **NYU**: Jason Phang, Thibault Févry, and Samuel R. Bowman, “Sentence Encoders on STILTs: Supplementary Training on Intermediate Labeled-Data Tasks,” ArXiv:1811.01088 [Cs], November 2, 2018, http://arxiv.org/abs/1811.01088
  - Typically extensive hyper-parameter search for fine-tuning:

Finetuning RoBERTa on Winograd Schema Challenge (WSC) data

The following instructions can be used to finetune RoBERTa on the WSC training data provided by SuperGLUE.

Note that there is high variance in the results. For our GLUE/SuperGLUE submission we swept over the learning rate (1e-5, 2e-5, 3e-5), batch size (16, 32, 64) and total number of updates (500, 1000, 2000, 3000), as well as the random seed. Out of ~100 runs we chose the best 7 models and ensembled them.

https://github.com/pytorch/fairseq/blob/master/examples/roberta/wsc/README.md

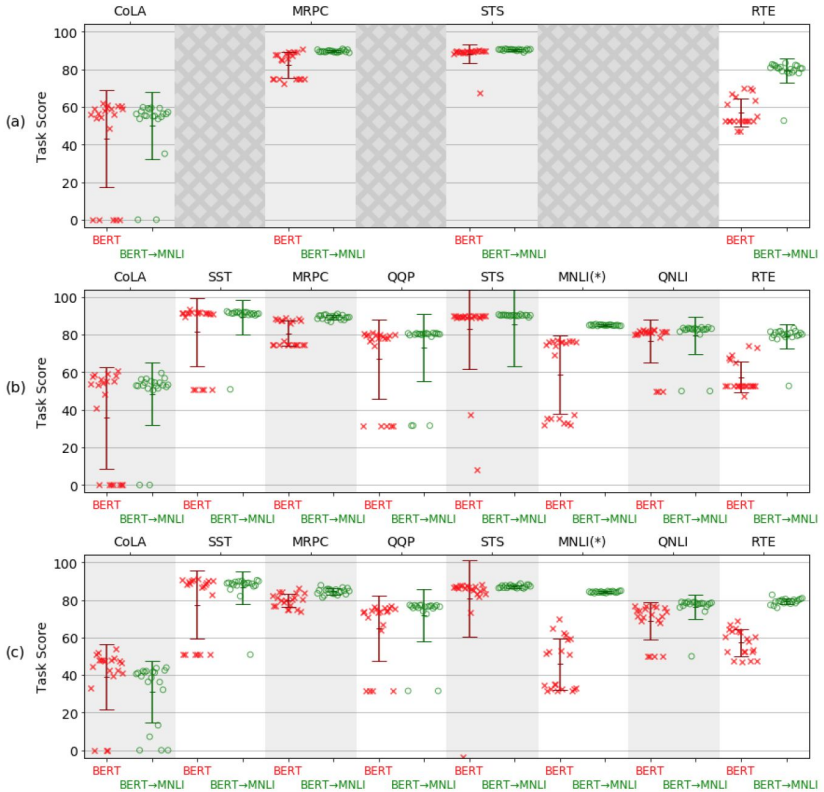
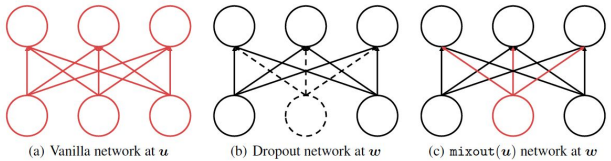


Figure 1: Distribution of task scores across 20 random restarts for BERT, and BERT with intermediary fine-tuning on MNLI. Each cross represents a single run. Error lines show mean±1std. (a) Fine-tuned on all data, for tasks with <10k training examples. (b) Fine-tuned on no more than 5k examples for each task. (c) Fine-tuned on no more than 1k examples for each task. (*) indicates that the intermediate task is the same as the target task.

Lack of robustness

What are our solutions?

- ❏ Better regularization?
  - ❏ **Mixout**: Cheolhyoung Lee, Kyunghyun Cho, and Wanmo Kang, "Mixout: Effective Regularization to Finetune Large-Scale Pretrained Language Models," ArXiv:1909.11299 [Cs, Stat], September 25, 2019, http://arxiv.org/abs/1909.11299.



(a) Vanilla network at u (b) Dropout network at w (c) $\texttt{mixout}(u)$ network at w

  - ❏ **Microsoft**: Haoming Jiang et al., "SMART: Robust and Efficient Fine-Tuning for Pre-Trained Natural Language Models through Principled Regularized Optimization," ArXiv:1911.03437 [Cs, Math], November 8, 2019, http://arxiv.org/abs/1911.03437.

$$\theta_{t+1} = \underset{\theta}{\operatorname{argmin}} \mathcal{F}(\theta) + \mu \mathcal{D}_{\text{Breg}}(\theta, \theta_t),$$

$$\min_{\theta} \mathcal{F}(\theta) = \mathcal{L}(\theta) + \lambda_s \mathcal{R}_s(\theta),$$

where $\mathcal{L}(\theta)$ is the loss function defined as

$$\mathcal{L}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \ell(f(x_i; \theta), y_i),$$

Regularizers on x and params

$$\mathcal{R}_s(\theta) = \frac{1}{n} \sum_{i=1}^{n} \max_{\|\widetilde{x}_i - x_i\|_p \leq \epsilon} \ell_s(f(x_i; \theta), f(\widetilde{x}_i; \theta))$$

$$\mathcal{D}_{\text{Breg}}(\theta, \theta_t) = \frac{1}{n} \sum_{i=1}^{n} \ell_s(f(x_i; \theta), f(x_i; \theta_t)$$

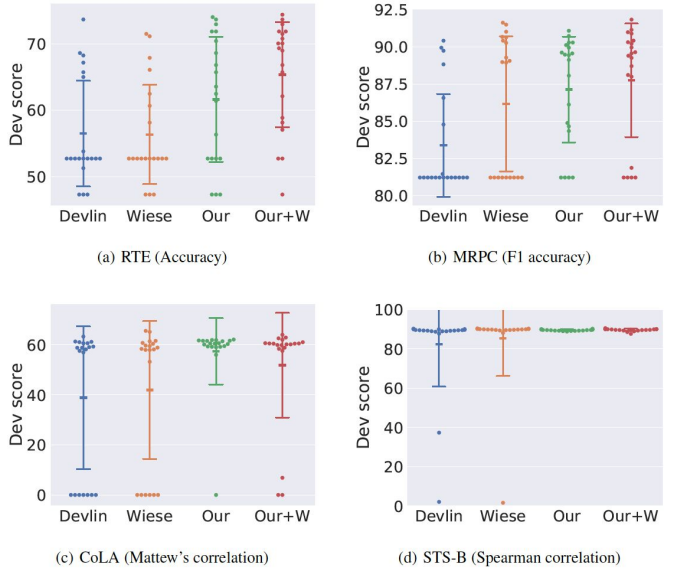


Figure 3: Distribution of dev scores on each task from 20 random restarts when finetuning $\text{BERT}_{\text{LARGE}}$ with Devlin et al. (2018)'s: both $\text{dropout}(0.1)$ and $\text{wdecay}(\mathbf{0},\ 0.01)$, Wiese et al. (2017)'s: $\text{wdecay}(\boldsymbol{w}_{\text{pre}},\ 0.01)$, ours: $\texttt{mixout}(\boldsymbol{w}_{\text{pre}},\ 0.7)$, and ours+Wiese et al. (2017)'s: both $\texttt{mixout}(\boldsymbol{w}_{\text{pre}},\ 0.7)$ and $\text{wdecay}(\boldsymbol{w}_{\text{pre}},\ 0.01)$. We write them as Devlin (blue), Wiese (orange), Our (green), and Our+W (red), respectively. Error intervals show mean±std. For all tasks, the number of finetuning runs that fail with the chance-level accuracy is significantly reduced when we use our regularization $\texttt{mixout}(\boldsymbol{w}_{\text{pre}},\ 0.7)$ regardless of using $\text{wdecay}(\boldsymbol{w}_{\text{pre}},\ 0.01)$.

Symmetrized KL divergence

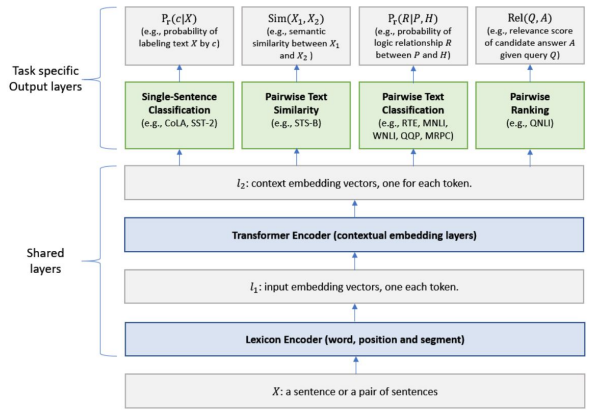
$$\ell_s(P, Q) = \mathcal{D}_{\text{KL}}(P||Q) + \mathcal{D}_{\text{KL}}(Q||P)$$

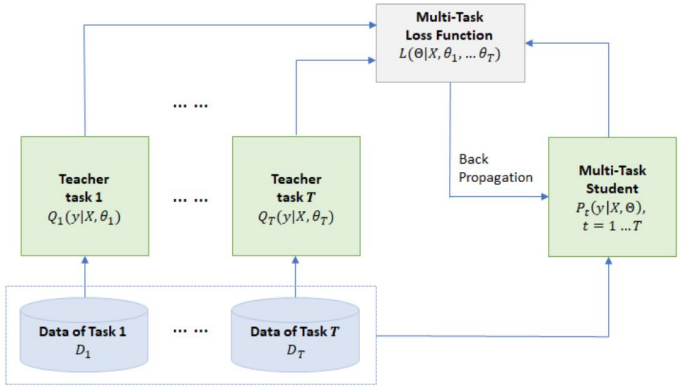
Lack of robustness

What are our solutions?

- Ensembles and multi-tasking
  - **Microsoft**: Xiaodong Liu et al., “Improving Multi-Task Deep Neural Networks via Knowledge Distillation for Natural Language Understanding,” ArXiv:1904.09482 [Cs], April 20, 2019, http://arxiv.org/abs/1904.09482

Multi-task learning + **ensembling** + **distillation**





Reporting and evaluation issues

- Current workflow for SOTA GLUE scores
- Comparing single runs on single splits
  - Show your work: asking people to report hyper-parameter searches
  - Reporting on standard splits leads to overfitting these splits
- Training and fine-tuning on various quantity of data
  - Debates on more data versus better models
  - How we solved the Winograd Schema Challenge

Reporting and evaluation issues

Typical workflow for fine-tuning to SOTA on GLUE

1. Pre-train your model with as much data/compute as possible
2. Tune fine-tuning hyperparameters on the dev sets
3. Use the SuperGLUE rather than GLUE data for WNLI and implement rescoring trick in combination with using additional labeled ("Definite Pronoun Resolution Dataset" http://www.hlt.utdallas.edu/~vince/data/emnlp12/) or unlabeled data (Vid Kocijan et al., "A Surprisingly Robust Trick for Winograd Schema Challenge," ACL 2019)
4. Use a special (and not officially allowed) pairwise ranking trick for QNLI and WNLI (users are not supposed to share information across test examples)
5. Intermediate MNLI task fine-tuning for MRPC/STS/RTE
6. Fine-tune many models on each task. Ensemble the best 5-10 models for each task.
7. Submit a (single) final run to the test leaderboard

Reporting and evaluation issues

Why is this not good

- ❑ Hyper-parameter search?

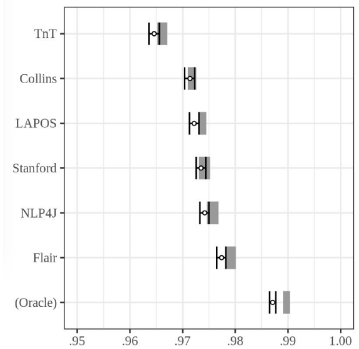
  Jesse Dodge et al., “Show Your Work: Improved Reporting of Experimental Results,” ArXiv:1909.03004 [Cs, Stat], September 6, 2019, http://arxiv.org/abs/1909.03004.

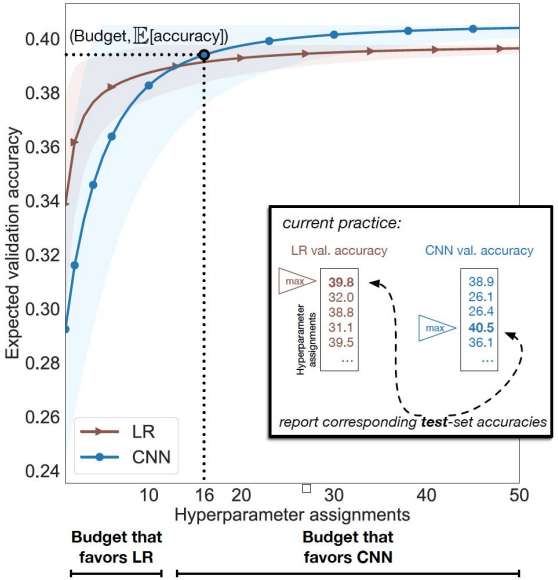
  $$\mathbb{E}\left[\max_{h\in\{h_1,\ldots,h_n\}} \mathcal{A}(\mathcal{M}, h, \mathcal{D}_T, \mathcal{D}_V) \mid n\right]$$

- ❑ “Standard” splits overfitting?

		PTB	ON
TnT	vs. Collins	20	20
Collins	vs. LAPOS	20	7
LAPOS	vs. Stanford	1	0
Stanford	vs. NLP4J	19	20
NLP4J	vs. Flair	20	20

Table 3: The number of random trials (out of twenty) for which the second system has significantly higher token accuracy than the first after Bonferroni correction. PTB, Penn Treebank; ON, OntoNotes.





- ✓ **For all reported experimental results**
  - ☐ Description of computing infrastructure
  - ☐ Average runtime for each approach
  - ☐ Details of train/validation/test splits
  - ☐ Corresponding validation performance for each reported test result
  - ☐ A link to implemented code
- ✓ **For experiments with hyperparameter search**
  - ☐ Bounds for each hyperparameter
  - ☐ Hyperparameter configurations for best-performing models
  - ☐ Number of hyperparameter search trials
  - ☐ The method of choosing hyperparameter values (e.g., uniform sampling, manual tuning, etc.) and the criterion used to select among them (e.g., accuracy)
  - ☐ Expected validation performance, as introduced in §3.1, or another measure of the mean and variance as a function of the number of hyperparameter trials.

Kyle Gorman and Steven Bedrick, “We Need to Talk about Standard Splits,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (ACL 2019, Florence, Italy: Association for Computational Linguistics, 2019), 2786–2791, https://doi.org/10.18653/v1/P19-1267.

More data or better models?

- More data for fine-tuning
  - How we solved the Winograd Schema Challenge?
- More data for pretraining
  - More data or better models – Debates on large-scale pretrained models (XLNet, RoBERTa…)
  - Scaling laws for neural LM
  - But transfer-learning => sample effectiveness?

More data or better models

Comparing models fine-tuned or pre-trained on different (quantity) of data

- **Finetuning**: solving the Winograd Schema Challenge
  - Winograd Schema Challenge
    The trophy would not fit in the brown suitcase because it was too big. What was too big? the trophy or the suitcase?
  - **MaskedWiki**: Kocijan, V., Cretu, A., Camburu, O., Yordanov, Y., & Lukasiewicz, T. (2019). A Surprisingly Robust Trick for the Winograd Schema Challenge. ACL.

MaskedWiki Dataset. To get more data for fine-tuning, we automatically generate a large-scale collection of sentences similar to WSC. More specifically, our procedure searches a large text corpus for sentences that contain (at least) two occurrences of the same noun. We mask the second occurrence of this noun with the [MASK] token. Several possible replacements for the masked token are given, for each noun in the sentence different from the replaced noun. We thus obtain examples that are structurally similar to those in WSC, although we cannot ensure that they fulfill all the requirements (see Section 2).

- **Pretraining**: more data versus better models
  - XLNet versus Bert debates
    https://medium.com/@xlnet.team/a-fair-comparison-study-of-xlnet-and-bert-with-large-models-5a4257f59dc0
  - RoBERTa versus XLNet
  - Then entered GPT2/T5/XLM-R/mBART – Scaling laws

Jared Kaplan et al., "Scaling Laws for Neural Language Models," *ArXiv:2001.08361 [Cs, Stat]*, January 22, 2020, http://arxiv.org/abs/2001.08361

More data or better models

Scaling laws for neural language models

- **Power** law of NLM **dataset/model/compute**

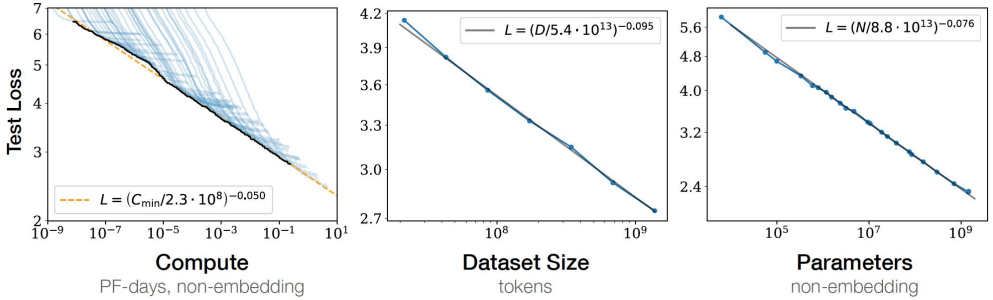


Figure 1 Language modeling performance improves smoothly as we increase the model size, datasetset size, and amount of compute[2] used for training. For optimal performance all three factors must be scaled up in tandem. Empirical performance has a power-law relationship with each individual factor when not bottlenecked by the other two.

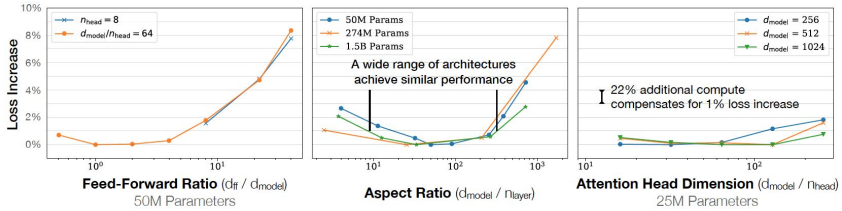
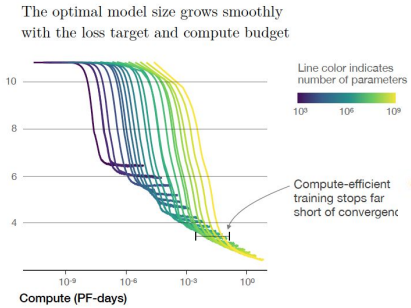
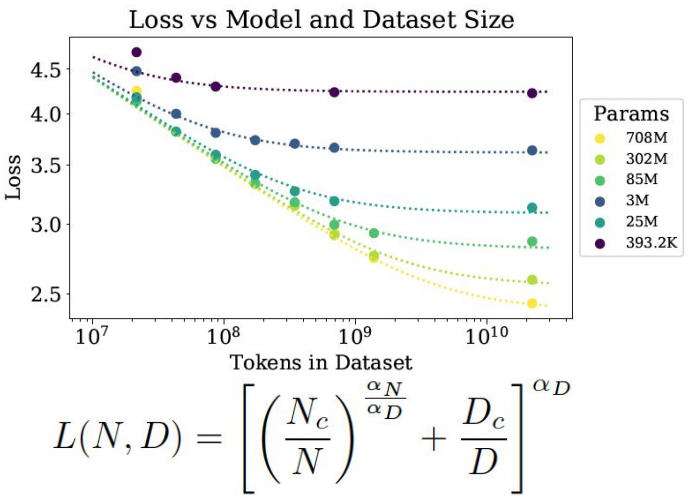
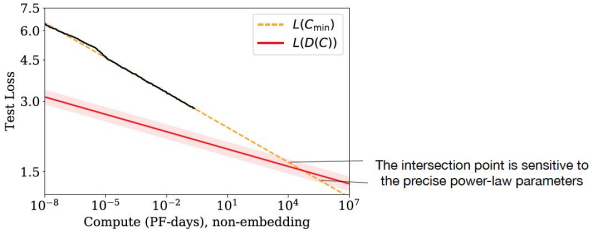


Figure 5 Performance depends very mildly on model shape when the total number of non-embedding parameters N is held fixed. The loss varies only a few percent over a wide range of shapes. Small differences





$$L(N, D) = \left[\left(\frac{N_c}{N}\right)^{\frac{\alpha_N}{\alpha_D}} + \frac{D_c}{D}\right]^{\alpha_D}$$



The intersection point of $L(D(C_{\min}))$ and $L(C_{\min})$

$C^* \sim 10^4$ PF-Days $N^* \sim 10^{12}$ parameters,

$D^* \sim 10^{12}$ tokens, $L^* \sim 1.7$ nats/token

More data or better models

Scaling laws for neural language models

- L – the cross entropy loss in nats. Typically it will be averaged over the tokens in a context, but in some cases we report the loss for specific tokens within the context.
- N – the number of model parameters, *excluding all vocabulary and positional embeddings*
- $C \approx 6NBS$ – an estimate of the total non-embedding training compute, where B is the batch size, and S is the number of training steps (ie parameter updates). We quote numerical values in PF-days, where one PF-day $= 10^{15} \times 24 \times 3600 = 8.64 \times 10^{19}$ floating point operations.
- D – the dataset size in tokens
- B_{crit} – the critical batch size [MKAT18], defined and discussed in Section 5.1. Training at the critical batch size provides a roughly optimal compromise between time and compute efficiency.
- C_{min} – an estimate of the minimum amount of non-embedding compute to reach a given value of the loss. This is the training compute that would be used if the model were trained at a batch size much less than the critical batch size.
- S_{min} – an estimate of the minimal number of training steps needed to reach a given value of the loss. This is also the number of training steps that would be used if the model were trained at a batch size much greater than the critical batch size.
- α_X – power-law exponents for the scaling of the loss as $L(X) \propto 1/X^{\alpha_X}$ where X can be any of $N, D, C, S, B, C^{\text{min}}$.

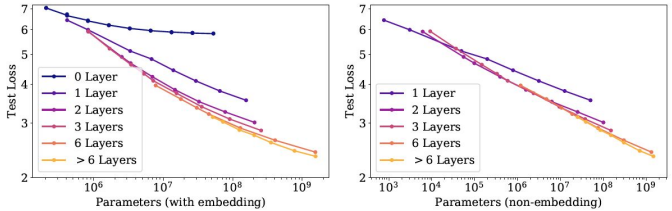


Figure 6 **Left:** When we include embedding parameters, performance appears to depend strongly on the number of layers in addition to the number of parameters. **Right:** When we exclude embedding parameters, the performance of models with different depths converge to a single trend. Only models with fewer than 2 layers or with extreme depth-to-width ratios deviate significantly from the trend.

Operation	Parameters	FLOPs per Token
Embed	$(n_{\text{vocab}} + n_{\text{ctx}})\, d_{\text{model}}$	$4d_{\text{model}}$
Attention: QKV	$n_{\text{layer}} d_{\text{model}} 3d_{\text{attn}}$	$2n_{\text{layer}} d_{\text{model}} 3d_{\text{attn}}$
Attention: Mask	—	$2n_{\text{layer}} n_{\text{ctx}} d_{\text{attn}}$
Attention: Project	$n_{\text{layer}} d_{\text{attn}} d_{\text{model}}$	$2n_{\text{layer}} d_{\text{attn}} d_{\text{embd}}$
Feedforward	$n_{\text{layer}} 2d_{\text{model}} d_{\text{ff}}$	$2n_{\text{layer}} 2d_{\text{model}} d_{\text{ff}}$
De-embed	—	$2d_{\text{model}} n_{\text{vocab}}$
Total (Non-Embedding)	$N = 2d_{\text{model}} n_{\text{layer}} (2d_{\text{attn}} + d_{\text{ff}})$	$C_{\text{forward}} = 2N + 2n_{\text{layer}} n_{\text{ctx}} d_{\text{attn}}$

Table 1 Parameter counts and compute (forward pass) estimates for a Transformer model. Sub-leading terms such as nonlinearities, biases, and layer normalization are omitted.

Yu, H., Edunov, S., Tian, Y., & Morcos, A.S. (2019). Playing the lottery with rewards and multiple languages: lottery tickets in RL and NLP. *ArXiv, abs/1906.02768*.

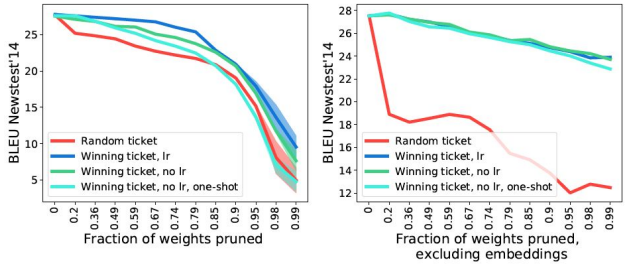


Figure 2: Winning ticket initialization performance for Transformer Base models trained on machine translation.

More data or better models

The question of generalization and data

Deep Mind: Dani Yogatama et al., "Learning and Evaluating General Linguistic Intelligence," *ArXiv:1901.11373 [Cs, Stat]*, January 31, 2019, http://arxiv.org/abs/1901.11373.

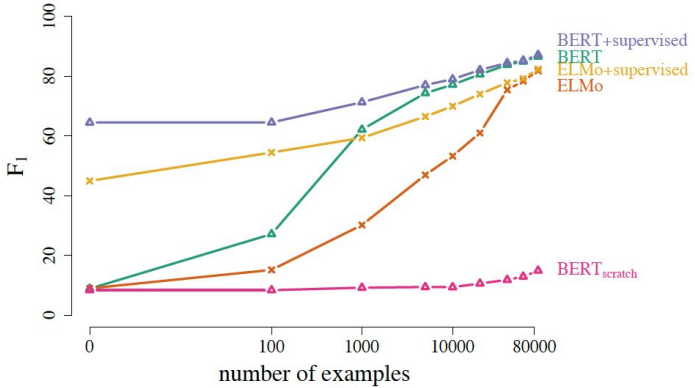
- Recent datasets **easy** to solve with little generalization or abstraction
  - gives models that only work well for a specific purpose
  - overestimates our success at having solved the general task
  - fails to reward sample efficient generalization
- Models typically evaluated in terms of performance at the **end** of training
  - model A: 90% accuracy with 100 training samples does not improve with more training
  - model B: takes one million examples to get to 90% before plateauing at 92%
- **Online code length**
  - **ENS**: Léonard Blier and Yann Ollivier, "The Description Length of Deep Learning Models," *ArXiv:1802.07044 [Cs]*, February 20, 2018, http://arxiv.org/abs/1802.07044
  - **DeepMind**: Dani Yogatama et al., "Learning and Evaluating General Linguistic Intelligence," *ArXiv:1901.11373 [Cs, Stat]*, January 31, 2019, http://arxiv.org/abs/1901.11373

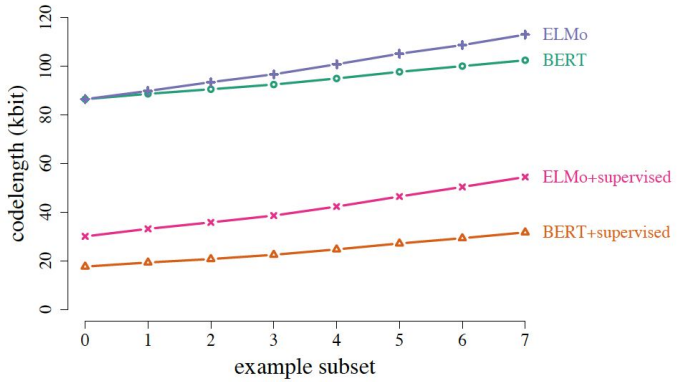
$$\ell(\mathcal{A}) = |\mathcal{S}_1| \log_2 |\mathcal{Y}| - \sum_{i=2}^{M} \log_2 p(y_{\mathcal{S}_i} \mid x_{\mathcal{S}_i}; \hat{\mathbf{W}}_{\mathcal{S}_{i-1}})$$

More data or better models

The question of generalization

- On SQuAD (and various QA datasets)





- Code-length metric: models that perform worse at the beginning can have problems **catching up** (catch-up phenomenon)
- One key reason models generalize poorly to new tasks is that they rely on **task specific components**

In-domain vs. out-of-domain generalization

Using more data and the question of in-domain versus out-of-domain

- In-domain generalization versus out-of-domain generalization

- What does out-of-domain generalization means?
  - train/test distribution shifts
    - In natural languages:
      - different training and test datasets for the same underlying “task”
      - designing new evaluation datasets
      - related to domain adaptation
      - related to zero-shot (but not exactly identical)
    - In artificially constructed languages
      - constructing different splits to evaluate performances under distributional shifts

In-domain vs. out-of-domain generalization

A few examples in NLP:

- **We've just seen an example on Question-Answering**

  On SQuAD: Dani Yogatama et al., "Learning and Evaluating General Linguistic Intelligence," *ArXiv:1901.11373 [Cs, Stat]*, January 31, 2019, http://arxiv.org/abs/1901.11373

	SQuAD	Trivia	QuAC	QA-SRL	QA-ZRE
BERT	86.5 (78.5)	35.6 (13.4)	56.2 (43.9)	77.5 (65.0)	55.3 (40.0)
ELMo	81.8 (72.2)	32.9 (12.6)	45.0 (34.5)	68.7 (52.3)	60.2 (42.0)

Table 2: F_1 (exact match) scores of the best BERT and ELMo models trained on SQuAD and evaluated on other question answering datasets.

- **On MNLI**: R. Thomas McCoy, Junghyun Min, and Tal Linzen, "BERTs of a Feather Do Not Generalize Together: Large Variability in Generalization across Models with Similar Test Set Performance," *ArXiv:1911.02969 [Cs]*, November 7, 2019, http://arxiv.org/abs/1911.02969

Heuristic	Definition	Example
Lexical overlap	Assume that a premise entails all hypotheses constructed from words in the premise	**The doctor** was **paid** by **the actor**. $\xrightarrow[\text{WRONG}]{}$ The doctor paid the actor.
Subsequence	Assume that a premise entails all of its contiguous subsequences.	The doctor near **the actor danced**. $\xrightarrow[\text{WRONG}]{}$ The actor danced.
Constituent	Assume that a premise entails all complete subtrees in its parse tree.	If **the artist slept**, the actor ran. $\xrightarrow[\text{WRONG}]{}$ The artist slept.

Figure 1: The heuristics targeted by the HANS dataset, along with examples of incorrect entailment predictions that these heuristics would lead to. (Figure from McCoy et al. (2019).)

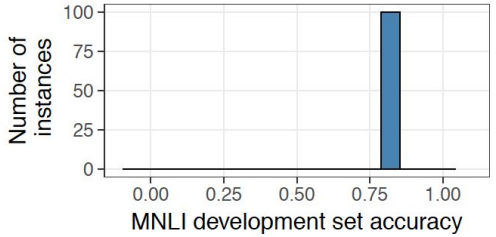


Figure 2: In-distribution generalization: Performance on the MNLI development set

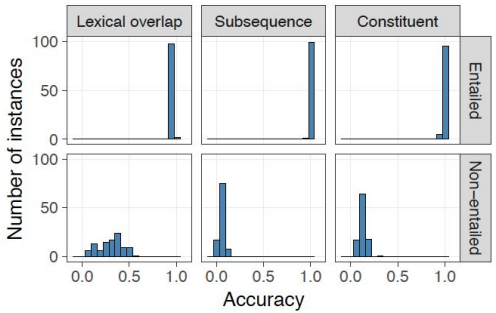


Figure 3: Out-of-distribution generalization: Performance on the HANS evaluation set, broken down into six categories of examples based on which syntactic heuristic each example targets and whether the correct label is *entailment* or *non-entailment*. The non-entailed lexical overlap cases (lower left plot) display a large degree of variability across instances.

In-domain vs. out-of-domain generalization

A few examples in NLP:

- **On MNLI**: R. Thomas McCoy, Junghyun Min, and Tal Linzen, "BERTs of a Feather Do Not Generalize Together: Large Variability in Generalization across Models with Similar Test Set Performance," *ArXiv:1911.02969 [Cs]*, November 7, 2019, http://arxiv.org/abs/1911.02969

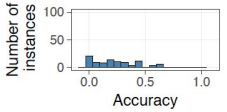
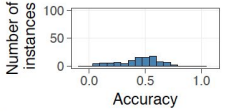
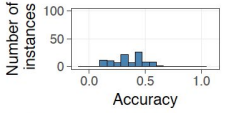
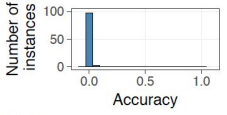
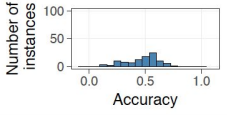
Category	Example	Accuracy distribution
Subject-object swap	The doctor visited the lawyer. ↛ The lawyer visited the doctor.	
Preposition	The judges by the manager saw the artists. ↛ The artists saw the manager.	
Relative clause	The actors advised the author who the tourists saw. ↛ The author advised the tourists.	
Passive	The senators were recommended by the managers. ↛ The senators recommended the managers.	
Conjunction	The doctors advised the presidents and the tourists. ↛ The presidents advised the tourists.	

Figure 4: Results on the various subcategories within the non-entailed lexical overlap examples of the HANS dataset. We do not include the other 25 subcategories of the HANS dataset in this figure as there was little variability across instances for those subcategories.

In-domain vs. out-of-domain generalization

A few examples in NLP:

- Work on compositionality: systematicity & generalization
  - **SCAN** (Brenden M. Lake and Marco Baroni, "Generalization without Systematicity: On the Compositional Skills of Sequence-to-Sequence Recurrent Networks," *ArXiv:1711.00350 [Cs]*, October 30, 2017, http://arxiv.org/abs/1711.00350)

jump	⇒	JUMP
jump left	⇒	LTURN JUMP
jump around right	⇒	RTURN JUMP RTURN JUMP RTURN JUMP RTURN JUMP
turn left twice	⇒	LTURN LTURN
jump thrice	⇒	JUMP JUMP JUMP
jump opposite left and walk thrice	⇒	LTURN LTURN JUMP WALK WALK WALK
jump opposite left after walk around left	⇒	LTURN WALK LTURN WALK LTURN WALK LTURN LTURN LTURN JUMP

Figure 1. Examples of SCAN commands (left) and the corresponding action sequences (right).

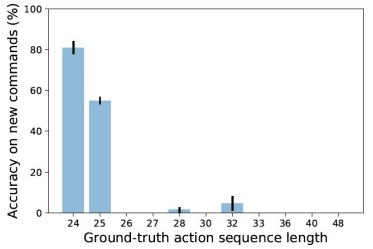
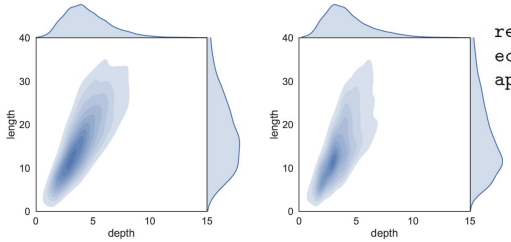


Figure 4. Zero-shot generalization to commands with action sequence lengths not seen in training. Top: accuracy distribution by action sequence length. Bottom: accuracy distribution by command length (only lengths attested in the test set shown, in both cases). Bars show means over 5 runs of overall-best model with ±1 SEM.

  - **PCFG SET** (Dieuwke Hupkes et al., "The Compositionality of Neural Networks: Integrating Symbolism and Connectionism," *ArXiv:1908.08351 [Cs, Stat]*, August 22, 2019, http://arxiv.org/abs/1908.08351)



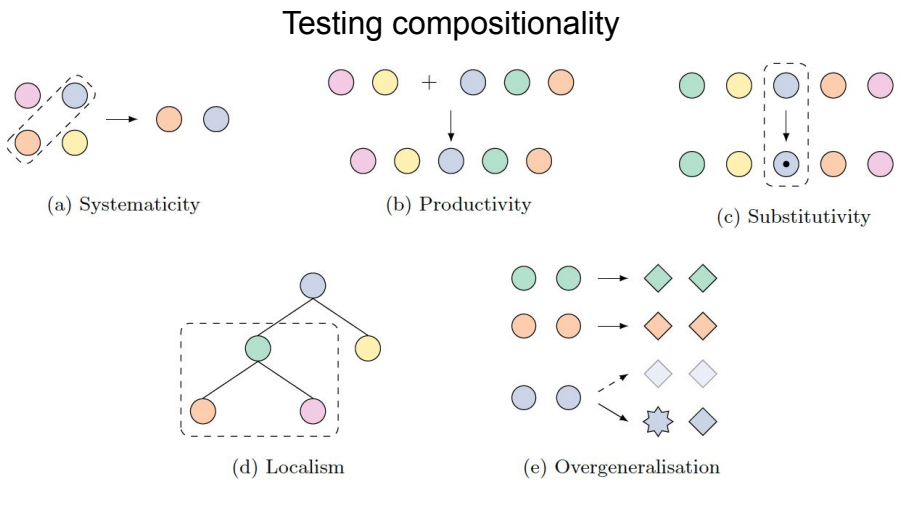
(a) WMT17 (b) PCFG SET data

```
repeat A B C                    →  A B C A B C
echo remove_first D , E F       →  E F F
append swap F G H ,| repeat I J  →  H G F I J I J
```

Unary functions F_U:		Binary functions F_B:	
copy $x_1 \cdots x_n$	$\rightarrow x_1 \cdots x_n$	append x, y	→ x y
reverse $x_1 \cdots x_n$	$\rightarrow x_n \cdots x_1$	prepend x, y	→ y x
shift $x_1 \cdots x_n$	$\rightarrow x_2 \cdots x_n x_1$	remove_first x, y	→ y
swap $x_1 \cdots x_n$	$\rightarrow x_n x_2 \cdots x_{n-1} x_1$	remove_second x, y	→ x
repeat $x_1 \cdots x_n$	$\rightarrow x_1 \cdots x_n x_1 \cdots x_n$		
echo $x_1 \cdots x_n$	$\rightarrow x_1 \cdots x_n x_n$		

In-domain vs. out-of-domain generalization

Dieuwke Hupkes et al., “The Compositionality of Neural Networks: Integrating Symbolism and Connectionism,” *ArXiv:1908.08351 [Cs, Stat]*, August 22, 2019, http://arxiv.org/abs/1908.08351



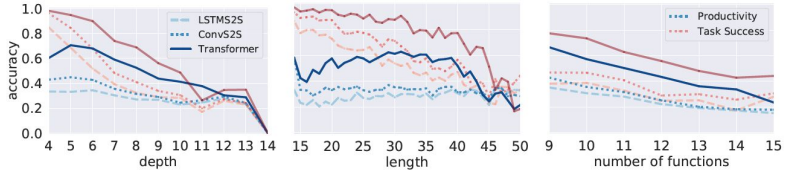


Figure 9: Accuracy of the three models on the productivity test set as a function of several properties of the input sequences: *depth* of the input's parse tree, the input sequence's *length* and the *number of functions* present. The results are averaged over three model runs and computed over ten thousand test samples.

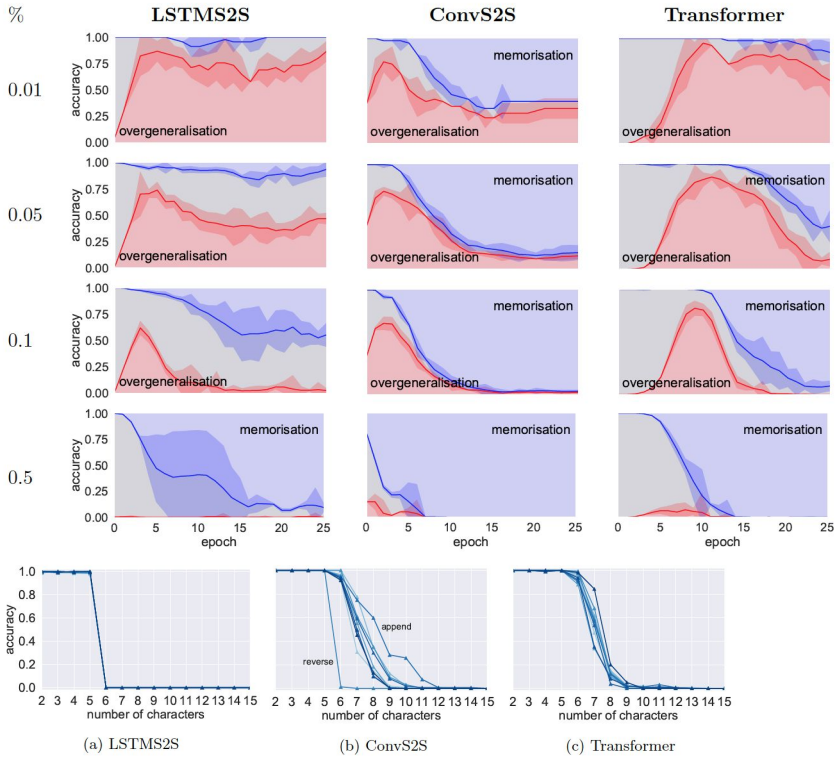


Figure 10: Accuracy of the three architectures on different functions with increasingly long character string inputs. The maximum character string length observed during training is 5. While Transformer and ConvS2S can, for most functions, generalise a little beyond this string length, LSTMS2S models cannot.

In-domain vs. out-of-domain generalization

Measuring train/test distribution shifts: large body of work in **domain adaptation**

- Plank, B., & Noord, G.V. (2011). Effective Measures of Domain Similarity for Parsing. *ACL*.
- Ruder, S., & Plank, B. (2017). Learning to select data for transfer learning with Bayesian Optimization. *EMNLP*.
- ElSahar, H., & Gallé, M. (2019). To Annotate or Not? Predicting Performance Drop under Domain Shift. *EMNLP/IJCNLP*.

- ❏ **Similarity metrics**: distance between the source and target domain
  - ❏ Kullback-Leibler (KL) divergence
  - ❏ Jensen-Shannon (JS) divergence
  - ❏ Renyi divergence
  - ❏ Maximum Mean Discrepancy (MMD)
  - ❏ Wasserstein distance
  - ❏ Proxy A distance
- ❏ **Feature Representations** for computing domain similarity measures
  - ❏ Term/n-grams distributions
  - ❏ Topic distributions (for instance by an LDA)
  - ❏ Word embeddings
  - ❏ Autoencoder representations
  - ❏ Token-sequence representations (diversity, n-grams)

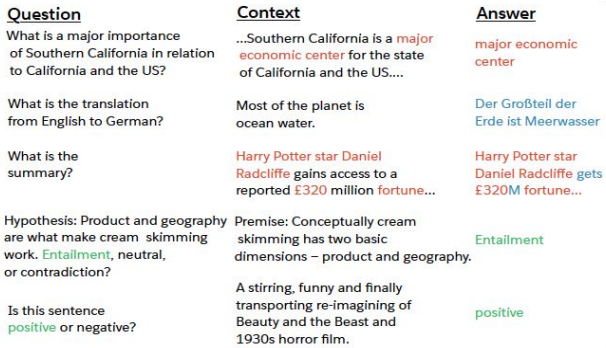
The limits of NLU and the rise of NLG

- Online code highlighted the question of training a task-specific head
  - should we even have task-specific elements?
- Welcome text-to-text models
  - GPT2 and language modeling as a multi-task learning objective
  - Facebook's BART and mBART: pretraining as text-to-text objective
  - Google's T5: finetuning as a text-to-text generation task
- NLU and NLG
  - Sam: nothing better than GLUE/SuperGLUE in the short-term
  - NLU and NLG - the problem of metrics
  - NeuralGen workshop

The limits of NLU and the rise of NLG

- Online code highlighted the question of training a task-specific head
- Welcome text-to-text models
  - The Natural Language Decathlon: getting rid of task-specific modules

    Bryan McCann et al., "The Natural Language Decathlon: Multitask Learning as Question Answering," *ArXiv:1806.08730 [Cs, Stat]*, June 20, 2018, http://arxiv.org/abs/1806.08730



Question	Context	Answer
What is a major importance of Southern California in relation to California and the US?	...Southern California is a major economic center for the state of California and the US....	major economic center
What is the translation from English to German?	Most of the planet is ocean water.	Der Großteil der Erde ist Meerwasser
What is the summary?	Harry Potter star Daniel Radcliffe gains access to a reported £320 million fortune...	Harry Potter star Daniel Radcliffe gets £320M fortune...
Hypothesis: Product and geography are what make cream skimming work. Entailment, neutral, or contradiction?	Premise: Conceptually cream skimming has two basic dimensions – product and geography.	Entailment
Is this sentence positive or negative?	A stirring, funny and finally transporting re-imagining of Beauty and the Beast and 1930s horror film.	positive

  - GPT2: language modeling as a multi-task learning objective

    Alec Radford et al., "Language Models Are Unsupervised Multitask Learners"

	LAMBADA (PPL)	LAMBADA (ACC)	CBT-CN (ACC)	CBT-NE (ACC)	WikiText2 (PPL)	PTB (PPL)	enwik8 (BPB)	text8 (BPC)	WikiText103 (PPL)	1BW (PPL)
SOTA	99.8	56.25	85.7	82.3	39.14	46.54	0.99	1.08	18.3	**21.8**
117M	**35.13**	45.99	**87.65**	**83.4**	**29.41**	65.85	1.16	1.17	37.50	75.20
345M	**15.60**	55.48	**92.35**	**87.1**	**22.76**	47.33	1.01	**1.06**	26.37	55.72
762M	**10.87**	**60.12**	**93.45**	**88.0**	**19.93**	**40.31**	**0.97**	**1.02**	22.05	44.575
1542M	**8.63**	**63.24**	**93.30**	**89.05**	**18.34**	**35.76**	**0.93**	**0.98**	**17.48**	42.16

Table 3. Zero-shot results on many datasets. No training or fine-tuning was performed for any of these results. PTB and WikiText-2 results are from (Gong et al., 2018). Other language model results are from (Dai et al., 2019).

"I'm not the cleverest man in the world, but like they say in French: **Je ne suis pas un imbecile [I'm not a fool].**

In a now-deleted post from Aug. 16, Soheil Eid, Tory candidate in the riding of Joliette, wrote in French: "**Mentez mentez, il en restera toujours quelque chose**," which translates as, "**Lie lie and something will always remain.**"

"I hate the word '**perfume**,"' Burr says. 'It's somewhat better in French: '**parfum**.'

If listened carefully at 29:55, a conversation can be heard between two guys in French: "**-Comment on fait pour aller de l'autre coté? -Quel autre coté?**", which means "**- How do you get to the other side? - What side?**".

If this sounds like a bit of a stretch, consider this question in French: **As-tu aller au cinéma?**, or **Did you go to the movies?**, which literally translates as Have-you to go to movies/theater?

"**Brevet Sans Garantie Du Gouvernement**", translated to English: "**Patented without government warranty**".

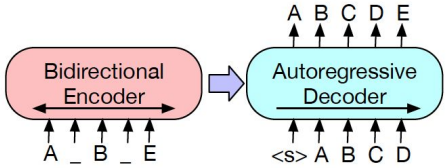
Table 1. Examples of naturally occurring demonstrations of English to French and French to English translation found throughout the WebText training set.

The limits of NLU and the rise of NLG

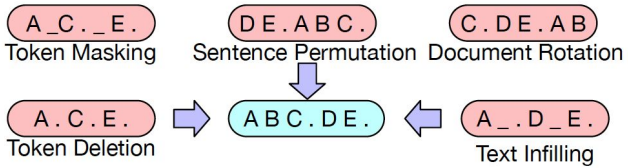
- The rise of pretrained NLG models
  - Facebook's BART: pretraining as text-to-text objective

    Mike Lewis et al., "BART: Denoising Sequence-to-Sequence Pre-Training for Natural Language Generation, Translation, and Comprehension," *ArXiv:1910.13461 [Cs, Stat]*, October 29, 2019, http://arxiv.org/abs/1910.13461.

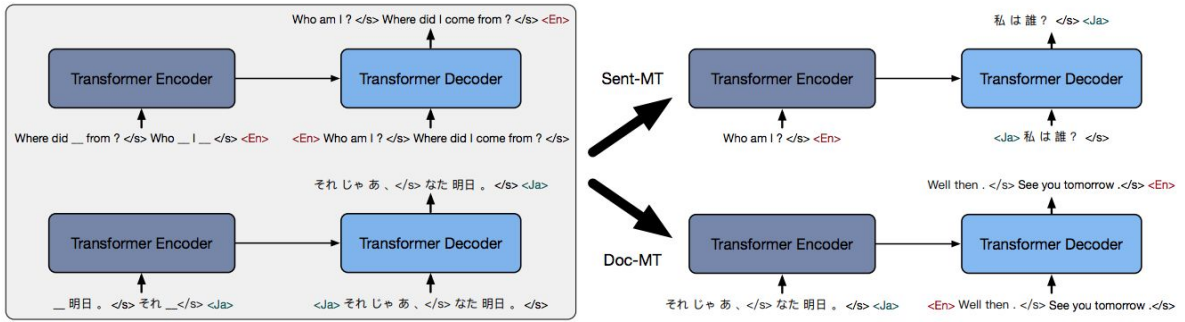
    Encoder-decoder scheme:



    Denoising objective:



  - mBART: Yinhan Liu et al., "Multilingual Denoising Pre-Training for Neural Machine Translation," *ArXiv:2001.08210 [Cs]*, January 23, 2020, http://arxiv.org/abs/2001.08210.

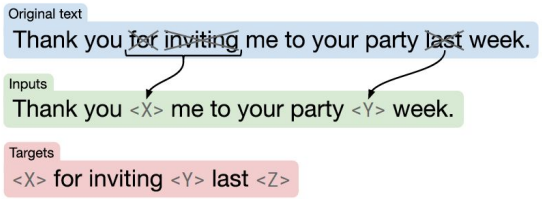


The limits of NLU and the rise of NLG

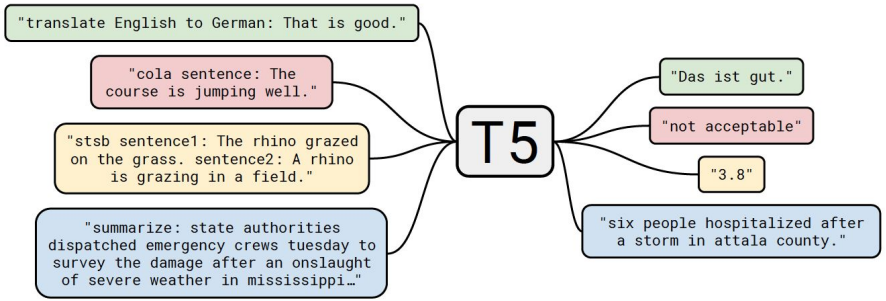
- The rise of pretrained NLG models
  - Google's T5: fine-tuning as a text-to-text generation task

Colin Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *ArXiv:1910.10683 [Cs, Stat]*, October 24, 2019, http://arxiv.org/abs/1910.10683.

Pretraining:



Fine-tuning:



GLUE: 90,3 (Human baseline: 87,1...)
SuperGLUE: 89,3 (Human baseline 89,8)

- **NLU and NLG**
  - Preparing a successor to GLUE and SuperGLUE?

The inductive bias question

- Let’s go back to the generalization problem
  - Models are brittle: fail when text is modified, even though its meaning is preserved
  - Models are spurious: memorize artifacts and biases instead of truly learning
- Out-of-domain generalization and inductive biases
- How should we formulate inductive bias
  - Linguistics tasks gives hints
  - Architectures: Graph Convolutional neural nets and Transformers
  - Let’s enrich our datasets

The inductive bias question

Let's go back to the generalization problem

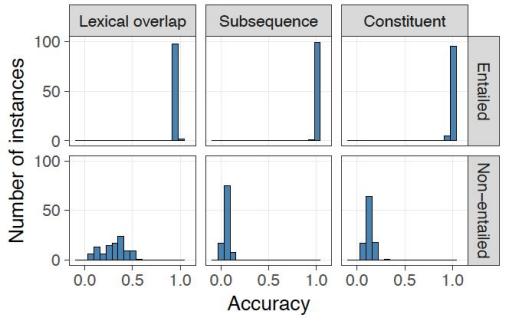
- ❏ Models are brittle: fail when text is modified, even with meaning preserved
- ❏ Models are spurious: memorize artifacts and biases instead of truly learning

Brittle

Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

Spurious

Heuristic	Definition	Example
Lexical overlap	Assume that a premise entails all hypotheses constructed from words in the premise	**The doctor** was **paid** by **the actor**. $\xrightarrow[\text{WRONG}]{}$ The doctor paid the actor.
Subsequence	Assume that a premise entails all of its contiguous subsequences.	The doctor near **the actor danced**. $\xrightarrow[\text{WRONG}]{}$ The actor danced.
Constituent	Assume that a premise entails all complete subtrees in its parse tree.	If **the artist slept**, the actor ran. $\xrightarrow[\text{WRONG}]{}$ The artist slept.

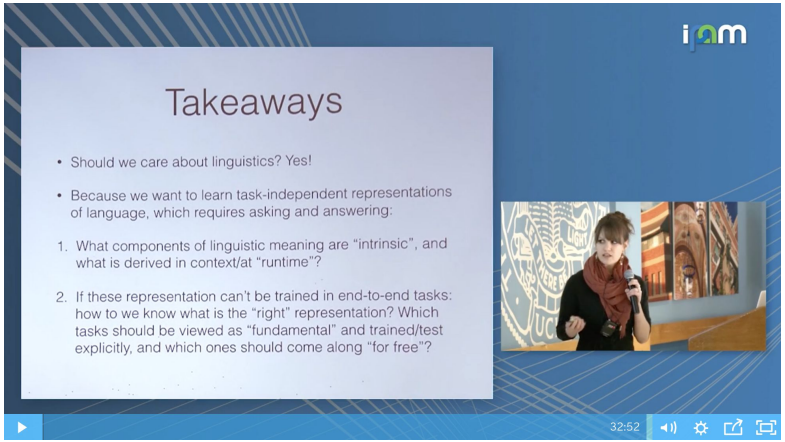


Robin Jia and Percy Liang, "Adversarial Examples for Evaluating Reading Comprehension Systems," ArXiv:1707.07328 [Cs], July 23, 2017, http://arxiv.org/abs/1707.07328

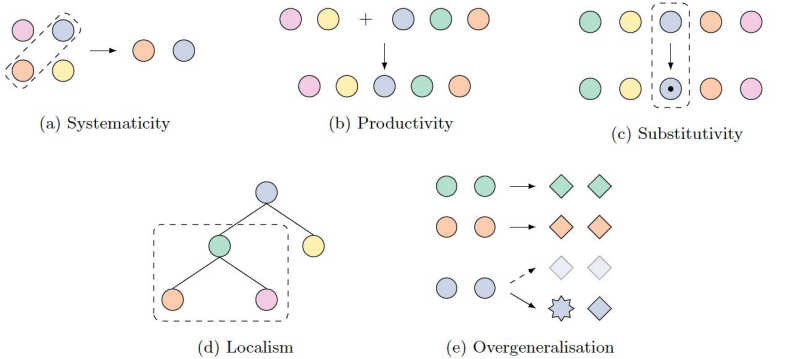
R. Thomas McCoy, Junghyun Min, and Tal Linzen, "BERTs of a Feather Do Not Generalize Together: Large Variability in Generalization across Models with Similar Test Set Performance," ArXiv:1911.02969 [Cs], November 7, 2019, http://arxiv.org/abs/1911.02969.

The inductive bias question

- A possible solution:
  - Providing better inductive bias in our models
- How should we test/design inductive bias
  - Linguistics!
    - Ellie Pavlick 2018 — Why should we care about linguistics
      http://www.ipam.ucla.edu/abstract/?tid=14546



- Dieuwke Hupkes et al., "The Compositionality of Neural Networks: Integrating Symbolism and Connectionism," *ArXiv:1908.08351 [Cs, Stat]*, August 22, 2019, http://arxiv.org/abs/1908.08351



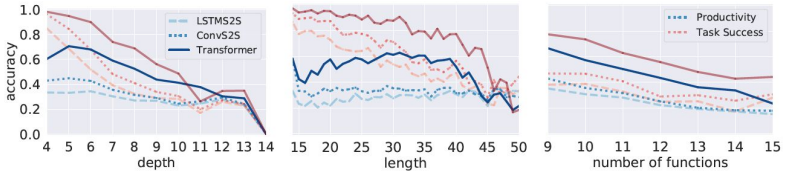


Figure 9: Accuracy of the three models on the productivity test set as a function of several properties of the input sequences: *depth* of the input's parse tree, the input sequence's *length* and the *number of functions* present. The results are averaged over three model runs and computed over ten thousand test samples.

The inductive bias question

- How should we formulate inductive bias
  - In the architectures:
    - With Graph Convolutional neural networks or Transformers

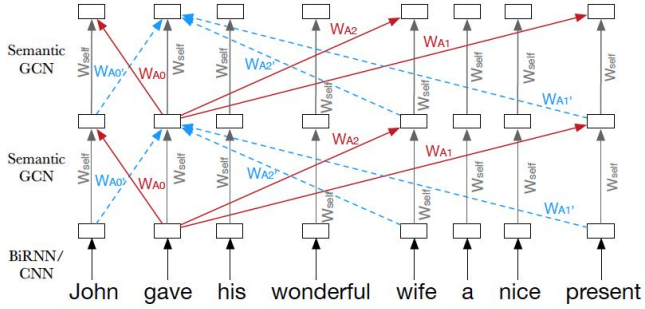
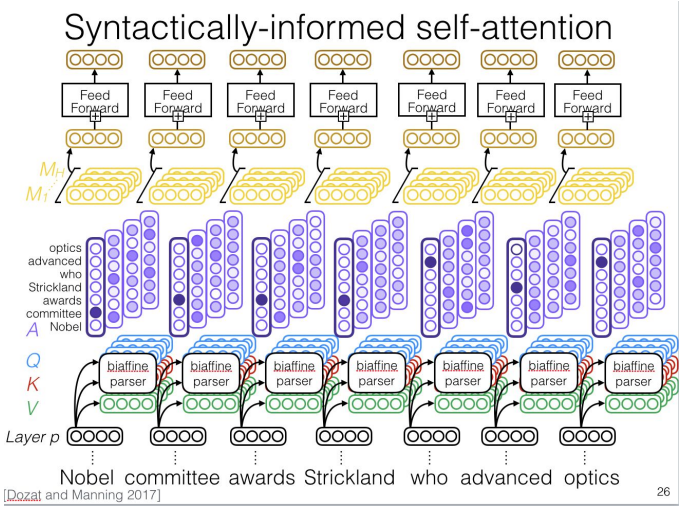


Figure 2: Two layers of semantic GCN on top of a (not shown) BiRNN or CNN encoder.



Diego Marcheggiani, Joost Bastings, and Ivan Titov, “Exploiting Semantics in Neural Machine Translation with Graph Convolutional Networks,” *ArXiv:1804.08313 [Cs]*, April 23, 2018, http://arxiv.org/abs/1804.08313

Emma Strubell et al., “Linguistically-Informed Self-Attention for Semantic Role Labeling,” *ArXiv:1804.08199 [Cs]*, April 22, 2018, http://arxiv.org/abs/1804.08199.

The inductive bias question

- A possible solution:
  - Providing better inductive bias in our models
- How should we formulate inductive bias
  - Enriching the data with inductive bias

"Overcoming the Lexical Overlap Bias Using Predicate-Argument Structures | OpenReview," accessed January 6, 2020, https://openreview.net/forum?id=2AGZUDRsHg

CGI: Adversarial SWAG

Premise: *A woman* is packing *a suitcase*.
Hypothesis: *A suitcase* is packing *a woman*.

Figure 3: Example of the GCI's syntactic variations set.

Premise: A lot of people are *sitting* on terraces in a big field and people is walking in the entrance of a big stadium.
Ending: A lot of people are *standing* on terraces in a big field and people is walking in the entrance of a big stadium.

Figure 4: Example of the GCI's antonym test set.

Premise: The reflection he sees is *Harrison Ford* as someone Solo winking back at him.
Ending: The reflection he sees is *Eve* as someone Solo winking back at him.

Figure 5: Example of the GCI's named entities test set.

Linguistically informed data augmentation

Original: Someone takes the drink, then holds it.
Augmented: Someone takes the drink, then holds it. `[PRD]` takes `[AG0]` Someone `[AG1]` the drink `[PRE]` `[PRD]` holds `[AG0]` Someone `[AG1]` it `[PRE]`

Figure 7: Augmenting the text of an input sentence with its predicate-argument structures.

Improved robustness

Evaluation	Model	Orig.	Aug.
In-domain	BERT	80.74	78.56
	RoBERTa	**83.22**	81.39
	XLNET	79.47	76.96
Syntactic	BERT	24.92	47.19
	RoBERTa	42.84	**57.98**
	XLNET	41.35	55.20
Antonym	BERT	15.33	35.10
	RoBERTa	29.83	**48.94**
	XLNET	27.98	42.34
Named Entities	BERT	7.93	15.87
	RoBERTa	21.16	**43.91**
	XLNET	19.57	40.21

The inductive bias question

Specialized pretraining tasks that teach what our model is missing

- Develop **specialized pretraining tasks** that explicitly learn such relationships
  - Word-pair relations that capture background knowledge (Joshi et al., NAACL 2019)
  - Span-level representations (Swayamdipta et al., EMNLP 2018)
  - Different pretrained word embeddings are helpful (Kiela et al., EMNLP 2018)
- Other pretraining tasks could explicitly learn **reasoning** or **understanding**
  - Arithmetic, temporal, causal, etc.; discourse, narrative, conversation, etc.
- Pretrained representations could be **connected in a sparse and modular way**
  - Based on linguistic substructures (Andreas et al., NAACL 2016) or experts (Shazeer et al., ICLR 2017)

The common-sense question

Models are brittle and spurious because they lack common-sense

- Limits of distributional hypothesis—difficult to learn certain types of information from raw text
  - Human reporting bias: not stating the obvious (Gordon and Van Durme, AKBC 2013)
  - Common sense isn't written down
  - Facts about named entities
  - No grounding to other modalities
- Possible solutions:
  - Incorporate other structured knowledge (e.g. knowledge bases like ERNIE, Zhang et al 2019)
  - Multimodal learning (e.g. with visual representations like VideoBERT, Sun et al. 2019)
  - Interactive/human-in-the-loop approaches (e.g. dialog, Hancock et al. 2018)

The common sense question

Definition of **Common Sense** (Yejin Choi's Talk at NeurIPS 2019 LIRE workshop)

- the basic level of practical knowledge and reasoning
- concerning everyday situations and events
- that are commonly shared among most people.

For example, it's ok to keep the closet door open, but it's not ok to keep the fridge door open, as the food inside might go bad.

Past failures (in 70s – 80s):

- weak computing power
- not much data
- no crowdsourcing
- not as strong computational models
- not ideal conceptualization / representations

The common sense question

7+ Commonsense Reasoning Challenges

1. Winogrande (AAAI 2020)
2. Physical IQA (AAAI 2020)
3. Social IQA (EMNLP 2019)
4. Cosmos QA (EMNLP 2019)
5. VCR: Visual Commonsense Reasoning (CVPR 2019)
6. Abductive Commonsense Reasoning (ICLR 2020)
7. TimeTravel: Counterfactual Reasoning (EMNLP 2019)
8. HellaSwag: Commonsense NLI (ACL 2019)

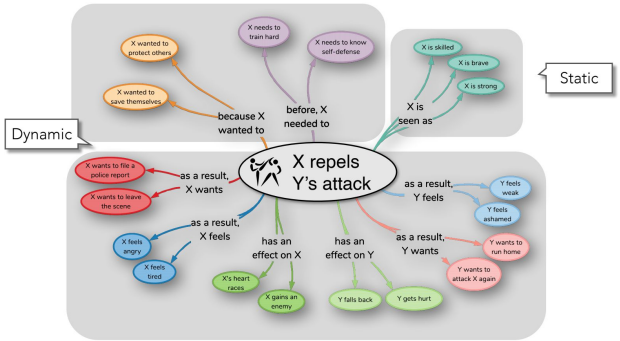
Important Observations:

Reasoning cannot be done using a finite set of inference rules over a finite set of variables

Yejin Choi's Talk at NeurIPS 2019 LIRE workshop

The common sense question

A few nice recent reads from Yeijin Choi's team:



- ## ATOMIC

  Maarten Sap et al., "ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning," *ArXiv:1811.00146 [Cs]*, February 7, 2019, http://arxiv.org/abs/1811.00146

- ## COMET

  Antoine Bosselut et al., "COMET: Commonsense Transformers for Automatic Knowledge Graph Construction," *ArXiv:1906.05317 [Cs]*, June 12, 2019, http://arxiv.org/abs/1906.05317.

- ## Winogrande

  Keisuke Sakaguchi et al., "WinoGrande: An Adversarial Winograd Schema Challenge at Scale," *ArXiv:1907.10641 [Cs]*, November 21, 2019, http://arxiv.org/abs/1907.10641.

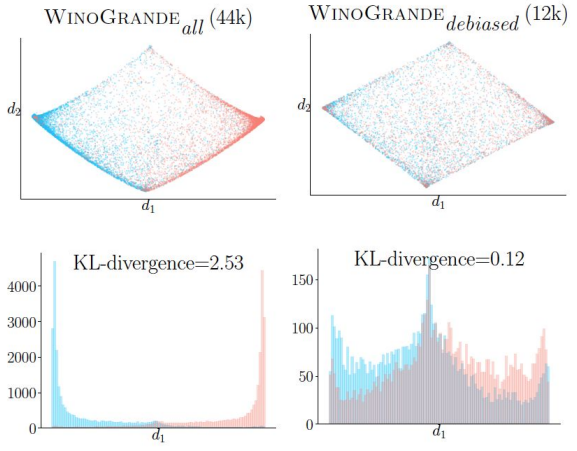
The common sense question

Winogrande

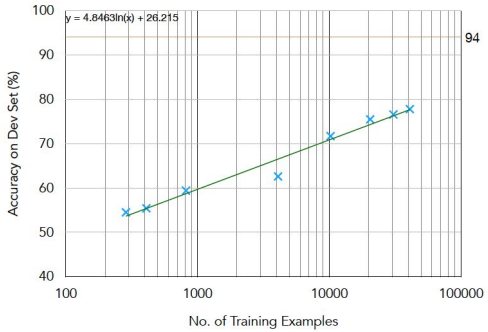
Keisuke Sakaguchi et al., “WinoGrande: An Adversarial Winograd Schema Challenge at Scale,” *ArXiv:1907.10641 [Cs]*, November 21, 2019, http://arxiv.org/abs/1907.10641

	Twin sentences	Options (**answer**)
✗	The monkey loved to play with the balls but ignored the blocks because he found **them** *exciting*.	**balls** / blocks
	The monkey loved to play with the balls but ignored the blocks because he found **them** *dull*.	balls / **blocks**
✗	William could only climb begginner walls while Jason climbed advanced ones because **he** was very *weak*.	**William** / Jason
	William could only climb begginner walls while Jason climbed advanced ones because **he** was very *strong*.	William / **Jason**
✓	Robert woke up at 9:00am while Samuel woke up at 6:00am, so **he** had *less* time to get ready for school.	**Robert** / Samuel
	Robert woke up at 9:00am while Samuel woke up at 6:00am, so **he** had *more* time to get ready for school.	Robert / **Samuel**
✓	The child was screaming after the baby bottle and toy fell. Since the child was *hungry*, **it** stopped his crying.	**baby bottle** / toy
	The child was screaming after the baby bottle and toy fell. Since the child was *full*, **it** stopped his crying.	baby bottle / **toy**

- Crowdsourcing:
  - Enhancing Crowd Creativity with random “anchor words” => 77k questions
  - Data Validation from crowd => 53k
- Light-weight adversarial filtering
  - fine-tune RoBERTa on 6k instances (removed from the dataset) => 46k
  - ensemble of linear classifiers (logistic regressions) trained on random subsets of the data determine whether the representation used in RoBERTa is strongly indicative of the correct answer => 13k questions (not all pairs)



Methods	dev acc. (%)	test acc.(%)
WKH	49.4	49.6
Ensemble LMs	53.0	50.9
BERT	65.8	64.9
RoBERTa	**79.3**	**79.1**
BERT (local context)	52.5	51.9
RoBERTa (local context)	52.1	50.0
BERT-DPR*	50.2	51.0
RoBERTa-DPR*	59.4	58.9
Human Perf.	94.1	94.0



Continual and meta-learning

- Current transfer learning **performs adaptation once**.
- Ultimately, we'd like to have models that continue to **retain and accumulate knowledge** across many tasks (Yogatama et al., 2019).
- No distinction between pretraining and adaptation; just **one stream of tasks**.
- Main challenge towards this: **Catastrophic forgetting**.
- Different approaches from the literature:
  - Memory, regularization, task-specific weights, etc.

Continual and meta-learning

- Objective of transfer learning: Learn a representation that is **general** and **useful** for many tasks.
- Objective **does not incentivize ease of adaptation** (often unstable); **does not learn how to adapt it**.
- **Meta-learning combined with transfer learning** could make this more feasible.
- However, most existing approaches are **restricted to the few-shot setting** and only **learn a few steps of adaptation**.

Bias

- Bias has been shown to be **pervasive in word embeddings and neural models** in general
- Large pretrained models **necessarily have their own sets of biases**
- There is a blurry boundary between common-sense and bias
- We need **ways to remove such biases** during adaptation
- A small fine-tuned model should be harder to misuse

Conclusion

- Themes: words-in-context, LM pretraining, deep models
- Pretraining gives better sample-efficiency, can be scaled up
- Predictive of certain features—depends how you look at it
- Performance trade-offs, from top-to-bottom
- Transfer learning is simple to implement, practically useful
- Still many shortcomings and open problems






That’s all for this year ;-)

Image credit: Andrejs Kirma