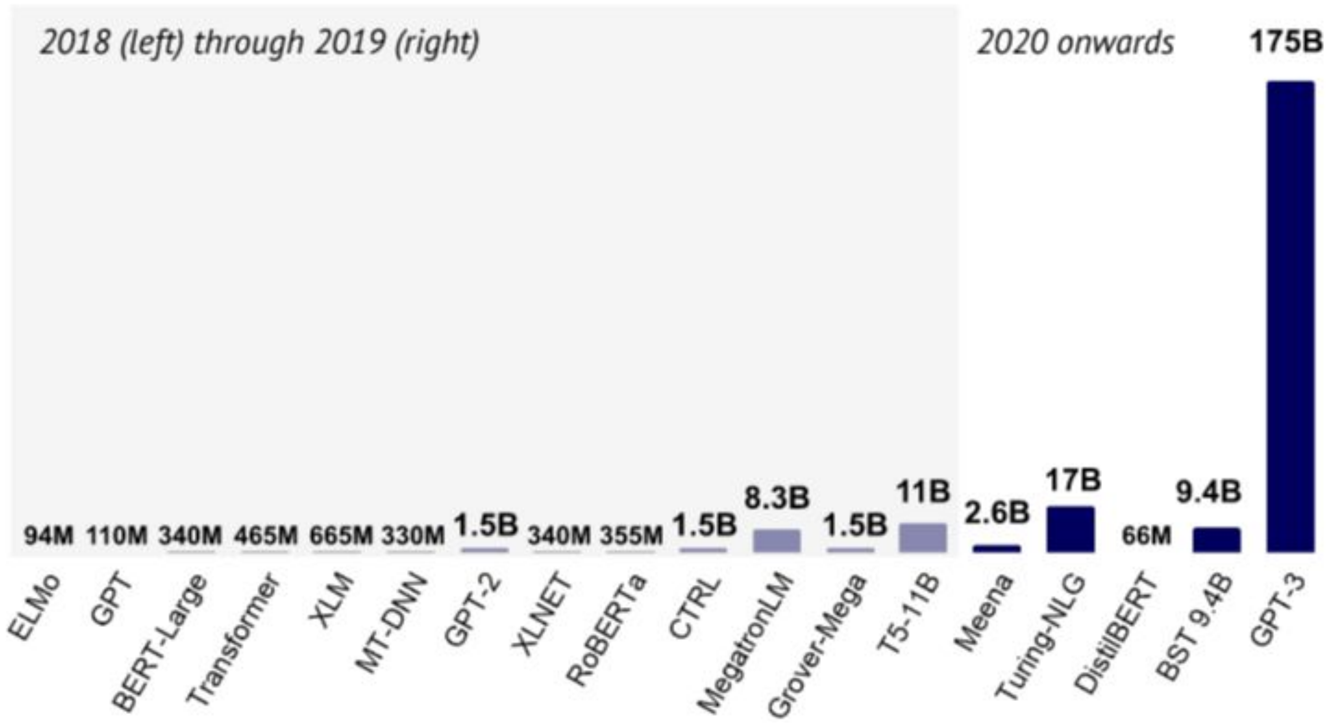
Advances in Sequence Knowledge Distillation

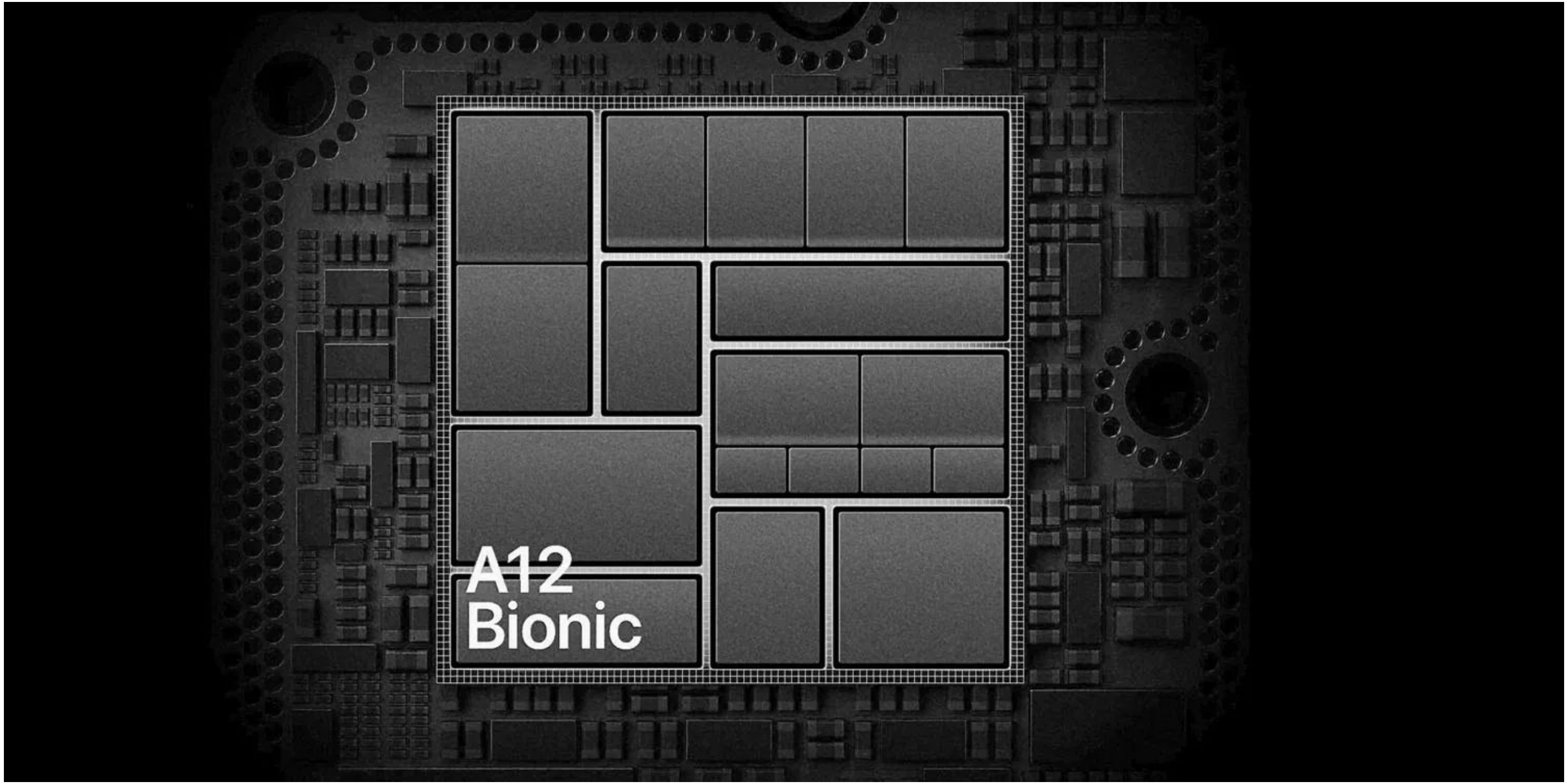
(with Yoon Kim, Demi Guo,
Sam Shleifer, and Victor Sanh)

NLP Models are Extremely Large

(State of AI, 2020)



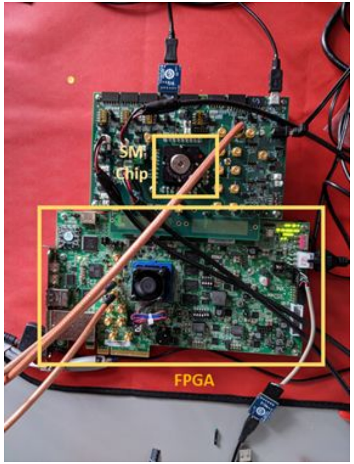
Edge Devices are Compute and Energy Limited



Research: Sustainability on the Edge

- Edge models that are fast and energy efficient

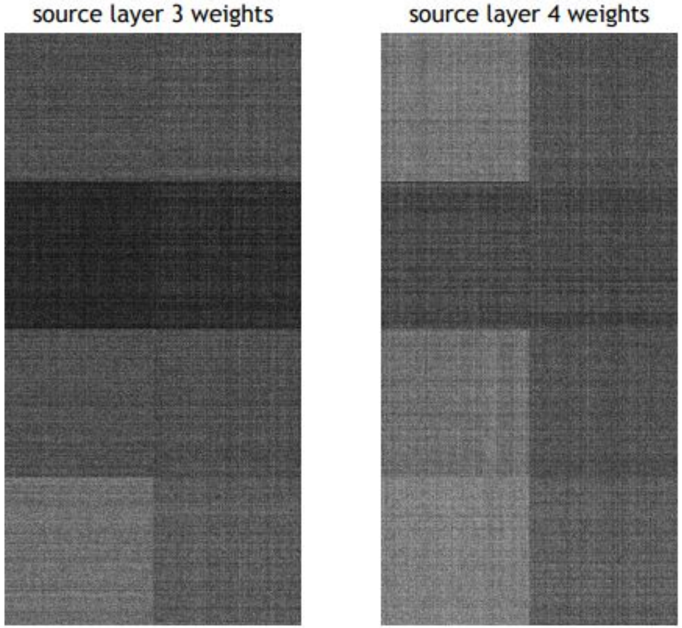
- Hardware co-design to run NLP systems



Many Options for NLP Compression

(See et al, 2016)

- Weight Pruning
- Quantization
- Early exit
- Layer drop
- Adapters
- ***Knowledge Distillation***
- ...



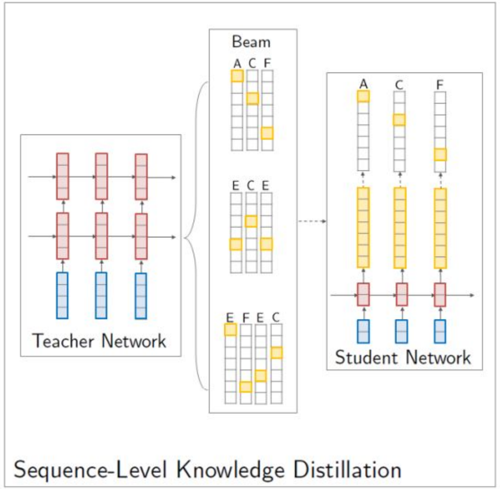
Knowledge Distillation is Particularly Appealing

Train small *student* to match larger *teacher*

- No constraints on final model structure
- Orthogonal to sparsity / quantization details
- Can ship directly to edge devices

Topic: Sequence Knowledge Distillation

- Sequence Knowledge Distillation (SeqKD)
- High-level: Learn *student* model by regenerating training data
- Effective compression for text generation e.g. MT, Summarization, NLG, …



Talk Overview

- Background: Knowledge Distillation
- Sequence KD: Challenges and Core Method
- Methodological Advances
- Applications Beyond Compression
- Research Suggestions

Background: Knowledge Distillation

Terminology: Knowledge Distillation

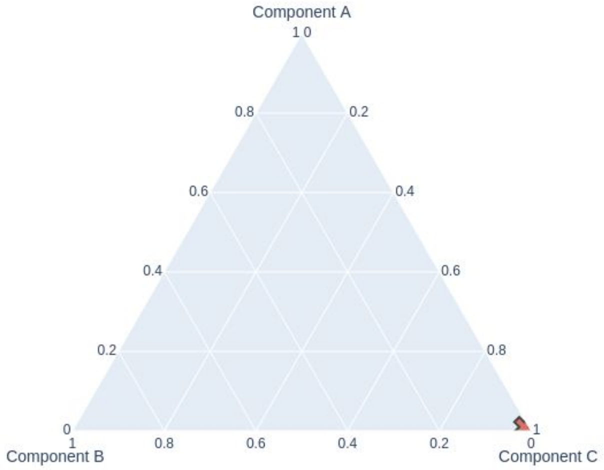
(Hinton et al, 2014)

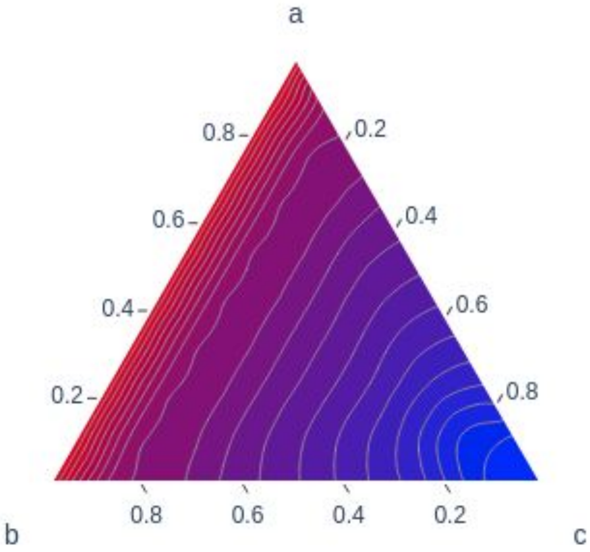
- Conditional Classification: Three classes (a, b, c)
    - Data points x, y
    - One-hot representation δ_y

- Two models: *teacher* and *student* (typically “smaller”)
    - Teacher predictions $\mathbf{p}_\theta = p(y \mid x \,; \theta)$
    - Student predictions $\mathbf{p}_\sigma = p(y \mid x \,; \sigma)$

Warmup: Standard MLE Training

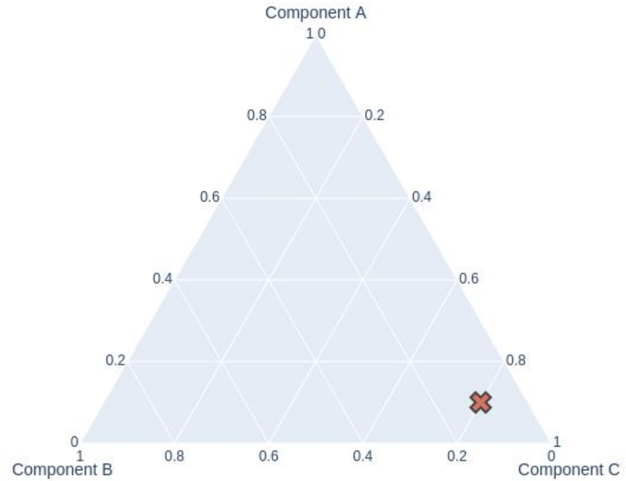
$$\arg\min_{\sigma} \sum_{x,y} \mathrm{KL}(\delta_y \;||\; \mathbf{p}_\sigma)$$

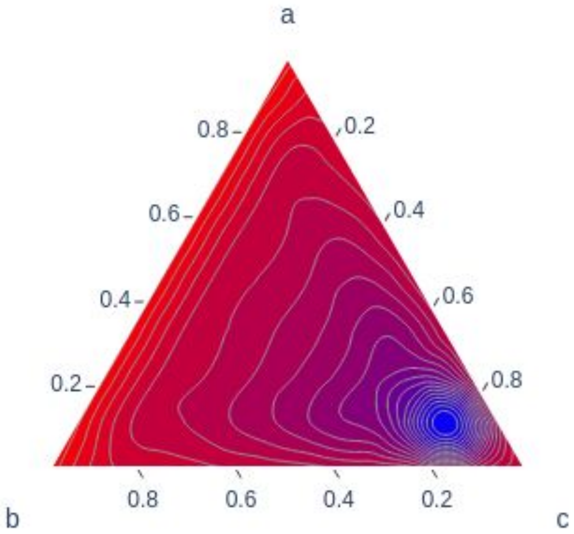




Warmup: Standard MLE with Label Smoothing

$$\arg\min_{\sigma} \sum_{x,y} \mathrm{KL}(\lambda\delta_y + (1-\lambda)\mathbf{u} \,\|\, \mathbf{p}_\sigma)$$

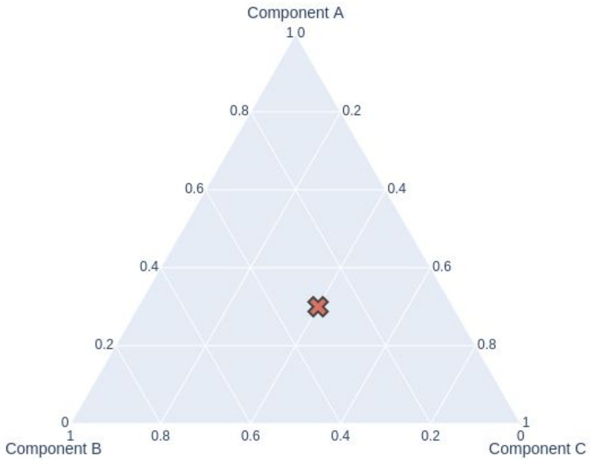


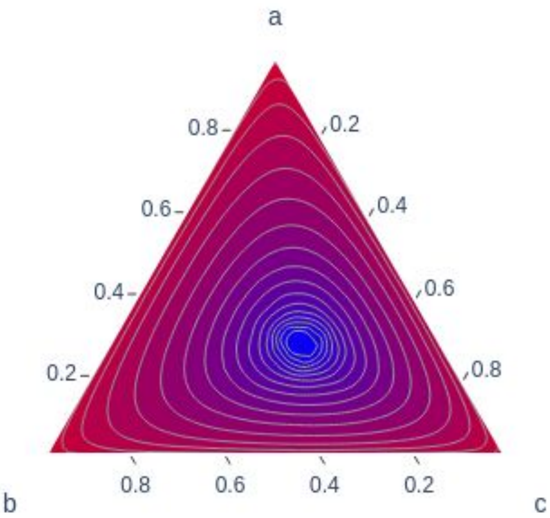


Knowledge Distillation

(Hinton et al, 2014)

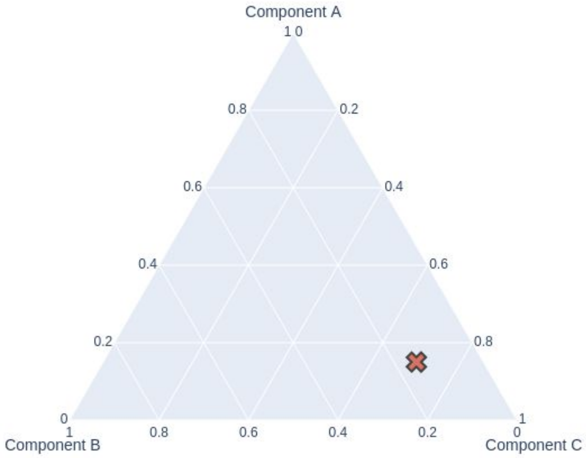
$$\arg\min_{\sigma} \sum_{x} \mathrm{KL}(\mathbf{p}_{\theta} \,||\, \mathbf{p}_{\sigma})$$

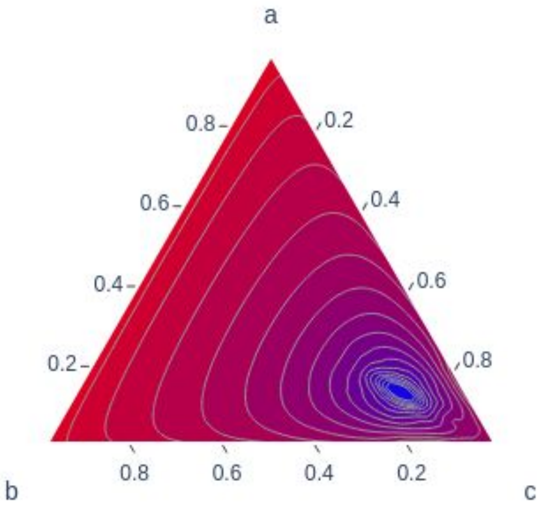




Knowledge Distillation + Soft Interpolation

$$\arg\min_{\sigma} \sum_{x,y} \mathrm{KL}(\lambda\delta_y + (1-\lambda)\mathbf{p}_\theta \parallel \mathbf{p}_\sigma)$$





Knowledge Distillation in NLP

- KD is a strong technique for classification benchmarks

- Many successful approaches for distilling BERT
  - DistilBERT, TinyBERT, MobileBERT, ...

- Additional techniques for transferring parameters and pre-distilling

Sequence KD:
Challenges and Core Method

Distillation for Generation

- Sequence generation tasks are different from classification
- Little success with KD at the token level
- Challenge: Sequential consistency with Teacher

Standard MLE Training with Autoregressive Model

$$\arg\min_{\sigma} \sum_{x,y} \mathrm{KL}(\delta_y \,||\, \mathbf{p}_\sigma)$$

$$=$$

$$\arg\min_{\sigma} \sum_{x,y} \sum_{i} \mathrm{KL}(\delta_y^{(i)} \,||\, \mathbf{p}_\sigma^{(i)})$$

MLE factors to local classification

Knowledge Distillation with Autoregressive Model

$$\arg\min_{\sigma} \sum_{x} \mathrm{KL}(\mathbf{p}_{\theta} \mid\mid \mathbf{p}_{\sigma})$$

$$\neq$$

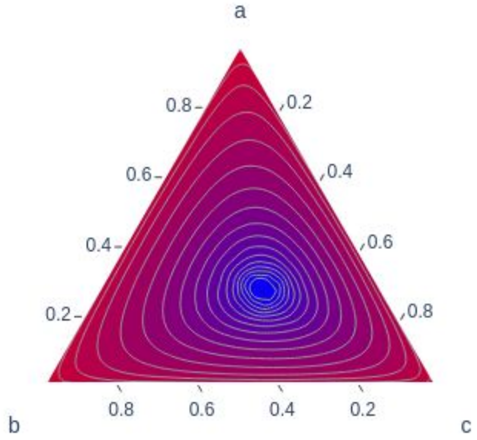
$$\arg\min_{\sigma} \sum_{x,y} \sum_{i} \mathrm{KL}(\mathbf{p}_{\theta}^{(i)} \mid\mid \mathbf{p}_{\sigma}^{(i)})$$

KD does not factors to local KD

Can we hope to compute this KL?

$$\arg\min_{\sigma} \sum_{x} \mathrm{KL}(\mathbf{p}_{\theta} \;||\; \mathbf{p}_{\sigma})$$

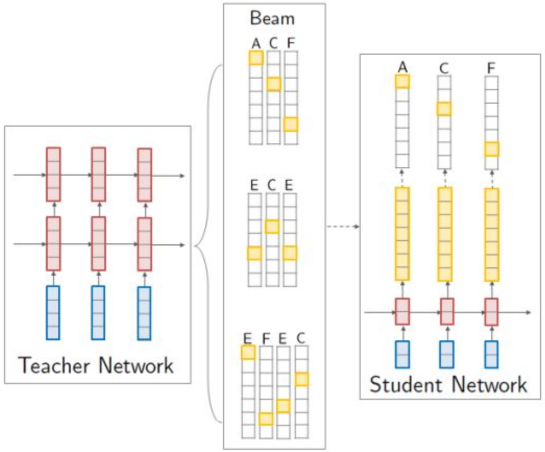
- Sum over sequences in a global model
- Exponential number of vertices ->



- Could approximate under assumption on teacher (see Struct Pred workshop)

Sequence Knowledge Distillation

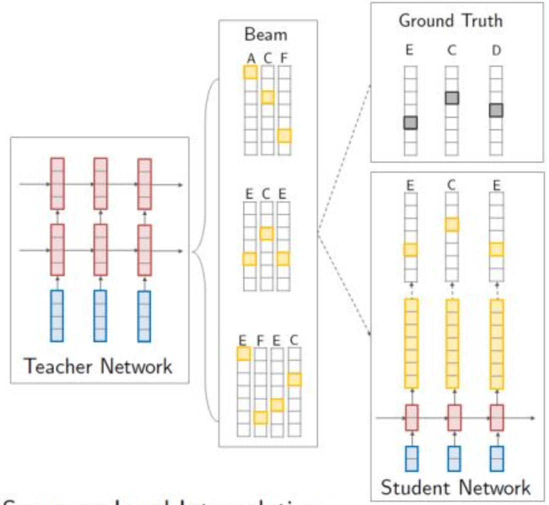
$$\arg\min_{\sigma} \sum_{x} \mathrm{KL}(\mathbf{p}_{\theta}^{*} \mid\mid \mathbf{p}_{\sigma})$$



Sequence-Level Knowledge Distillation

Sequence Knowledge Distillation + Interpolation

$$\arg\min_{\sigma} \sum_{x,y} \mathrm{KL}((\lambda\delta_y + (1-\lambda)\mathbf{p}_\theta)^* \;||\; \mathbf{p}_\sigma)$$



Sequence-Level Interpolation

Sequence Knowledge Example

Original (x): Bis 15 Tage vor Anreise sind Zimmer-Annullationen kostenlos

Original (y): Room cancellation is free up to 15 days before arrival

SeqKD (p^*): Up to 15 days prior to arrival it is free

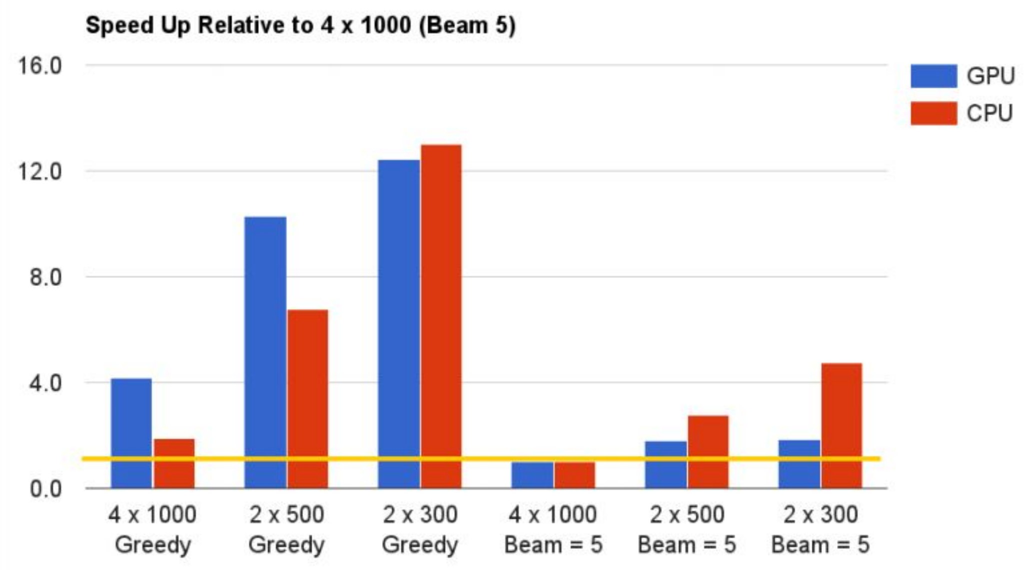
SeqInter ((y + p)^*) : Up to 15 days prior to arrival <unk> is free

- Does not always make it *better* but tends to be more direct.

Original Results

- Low absolute performance in retrospect (5 years ago)
- Relative results show >> KD, and major practical speed-ups

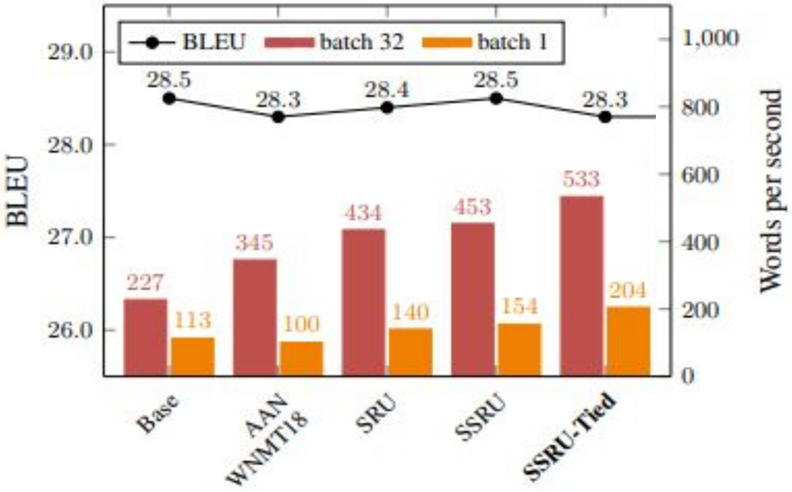
Relative
Speed Up



Teacher

Ludicrously Fast Neural Machine Translation (Kim et al, 2019)

- WNMT Efficiency Task
- Fancy SeqKD from Transformer to an fast RNN (SRU) model



SeqKD For Accuracy. MSR Asia at WMT 2019

- Pipeline of noisy data augmentation techniques for increasing accuracy
- “We iterate back translation and knowledge distillation multiple times to boost the performance of the model”
- Related to Born-Again Networks, repeated distillation

Why Might Distillation Improve Accuracy?

- Self-Training for Structured Prediction (McClosky et al, 2006)
  - Semi-supervised learning by training on automatically labeled data
  - Iteratively label data and retrain

- Hope and Fear Translation Training (Chiang, 2012)
  - MT training is often too hard
  - Use a model to adjust training to loss based on difficulty

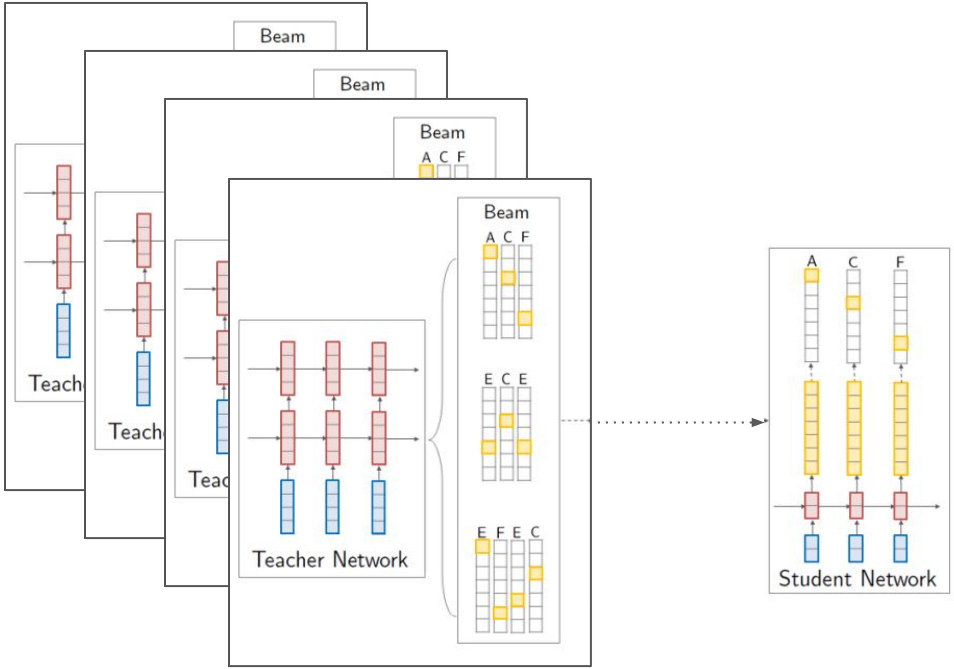
Extensions and Advances

Ensemble Distillation

(Freitag et al, 2017)

- Goal: Learn a student with the accuracy of an ensemble

- Significant performance benefits for little inference cost

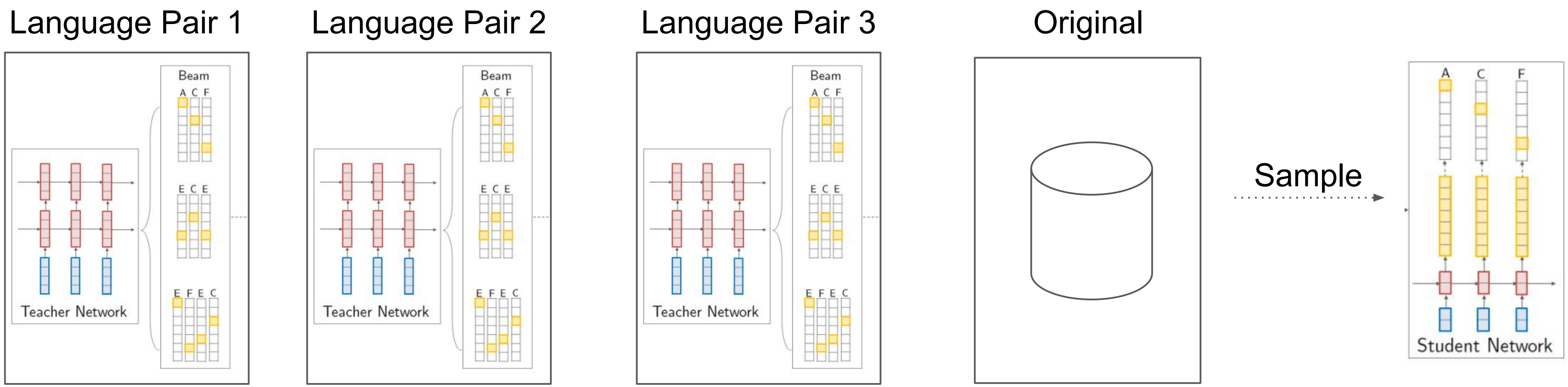


Multilingual Translation

(Tan et al, 2019)

- Goal: Train a single multilingual model from many single pairs

- Approach: Mimic each teacher in turn, *if* student is still worse



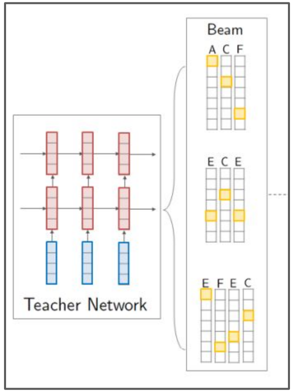
Domain Adaptation

(Dakwale and Monz, 2017)

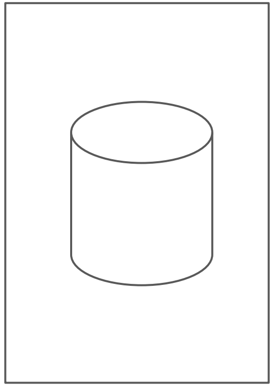
- Goal: Train an in-domain student model that doesn't forget how to translate

- Approach: Alternate between domain fine-tuning and out-of-domain seqKD

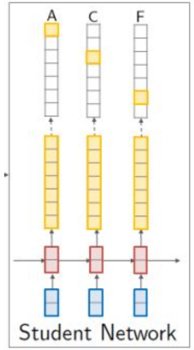
Teacher (OOD)



In-Domain Data



Sample



Application: Model Stealing

- Goal: Imitate a blackbox production model to make adversarial attacks

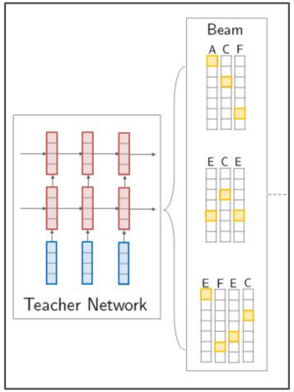
- Utilizes seqKD style approaches to imitate models

Test	Model	Google	Bing	Systran
WMT	Official	32.0	32.9	27.8
	Imitation	31.5	32.4	27.6
IWSLT	Official	32.0	32.7	32.0
	Imitation	31.1	32.0	31.4

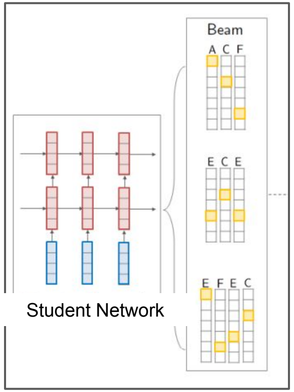
Can we do better? Imitation Learning (Lin et al, EMNLP 2020)

- DAgger imitation distillation algorithm
- Intuition: Explores more of distribution based on student exploration

Teacher Sampled Sequences

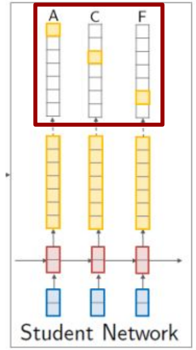


Student Sampled Sequences



Sample

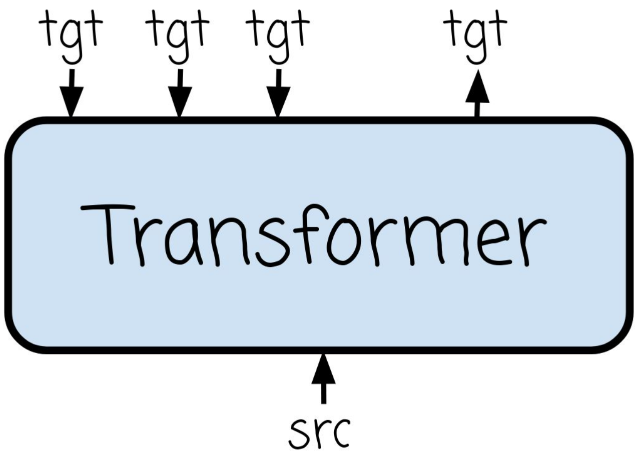
Teacher Oracle Labels



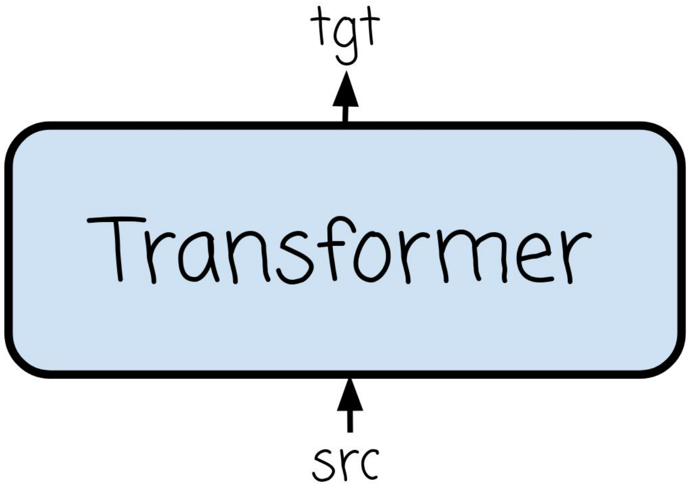
Further Edge Applications

Non-Autoregressive Translation

(Gu et al, 2018)



Autoregressive



Non-Autoregressive

Non-Autoregressive Translation

(Gu et al, 2018)

- “We see that training on the distillation corpus rather than the ground truth provides a fairly consistent improvement of around 5 BLEU points.”

- Still necessary in recent NAT systems (Deng and Rush, 2020)

Distillation

WMT14	
En-De	De-En
27.41	31.49
26.34	30.69
26.43	30.72
26.52	30.73
26.80	31.22
26.90	31.15
26.92	31.23

No Distillation

WMT14	
En-De	De-En
27.41	31.49
21.34	26.91
22.55	27.56
23.09	27.79
23.35	28.64
24.40	29.43

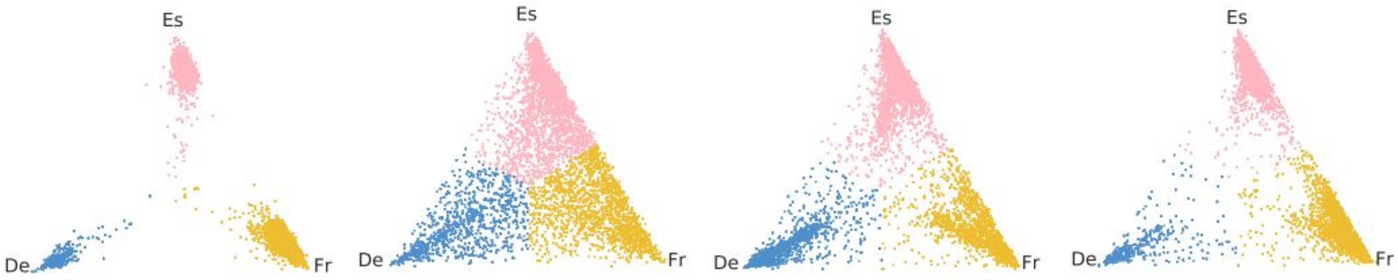
Faster

Slower

Why does this work? Unimodality

(Zhou et al, 2020)

- Synthetic dataset where each src is paired with 3 outputs in different languages
- Hypothesis: NAT cannot fix its mode, will output words from all 3 languages
- Sequence Distillation from teacher fixes a mode of each input sentence



(a) AT Baseline (b) NAT Baseline (c) NAT Random Select (d) NAT Distill

Application: Text-To-Speech

(Ren et al, 2019)

- FastSpeech / WaveGlow - 2019

- Trains a model analogous to non-autoregressive MT for generation

System	CMOS
FastSpeech	0
FastSpeech without 1D convolution in FFT block	-0.113
FastSpeech without sequence-level knowledge distillation	-0.325

Table 4: CMOS comparison in the ablation studies.

Pretraining and Sequence Distillation? (Shleifer and Rush, 2020)

- Pretrained sequence models (BART, PEGASUS, ...) show impressive results

- Early results show that seqKD may not be necessary c

Teacher	Size	Data	Teacher Score	Shrink Score	Shrink Cost	KD Score	KD Cost	SeqKD Score	SeqKD Cost
BART †	12-3	XSUM	22.29	21.08	2.5	**21.63**	6	21.38	15
Pegasus	16-4	XSUM	24.56	22.64	13	21.92	22	**23.18**	34
BART	12-6	CNN	21.06	**21.21**	2	20.95	14	19.93	19.5
Pegasus	16-4	CNN	21.37	**21.29**	31	-	-	20.1	48
Marian	6-3	EN-RO	27.69	25.91	4	24.96	4	**26.85**	28
mBART	12-3	EN-RO	26.457	25.6083	16	25.87	24	**26.09**	50

Open Questions

Sequence Distillation is more than Compression

- Distillation for ensembling
- Distillation for model transfer
- Distillation for model stealing

Many other possibilities...

- Distillation for specialized architectures
- Distillation for energy use
- Distillation for more efficient pretraining?

Simplification

- Results show that distillation is removing ambiguity and complexity
- Is this a positive thing? Or is it just learning to exploit metrics?
- How can we better measure what is being lost?

Distillation + Compression X

- Results hint that distillation is partially orthogonal to other compression
- However, distilled models tend to be more confident and less stable
- Unclear how to tell when they are “too” compressed

Conclusions

- Sequence distillation is interesting, with many NLP specific challenges

- More work to be done to decide how (and if) it works

- Basic approach remains easy to do and surprisingly broad in use