NMT DISTILLATION

By: Sergio Charles

With: Eric & Oleksii

Special Thanks: Som & Sandeep

August 31, 2021

NVIDIA

OUTLINE

- **Background & Motivation**
- Hinton-style Distillation
  - ModelPT Distillation
- DistilBERT-style Distillation
- Sequence-Level Distillation
  - Greedy-search and beam-search sampling
- Hybrid Distillation
- Distillation in the Low Data Regime
- NMT Distillation Recommendations
- Future Work

NMT MODEL ARCHITECTURE

Attentional seq-to-seq encoder-decoder

- Models $p(\mathbf{y}|\mathbf{x})$ with source sentence $\mathbf{x} = [\mathbf{x_1}, \ldots, \mathbf{x}_{|\mathcal{S}|}]$ and target sentence $\mathbf{y} = [\mathbf{y_1}, \ldots, \mathbf{y}_{|\mathcal{T}|}]$
- Encoder: transforms **x** into continuous representations (e.g. Bi-drectional RNN, Transformer)
- Decoder: predict conditional distribution of each target word using beam search, conditioned on encoding
- Machine translation seeks to solve:

$$\text{argmax}_{\mathbf{y}\in\mathcal{T}} p(\mathbf{y}|\mathbf{x})$$

NMT MODEL ARCHITECTURE

Attentional seq-to-seq encoder-decoder

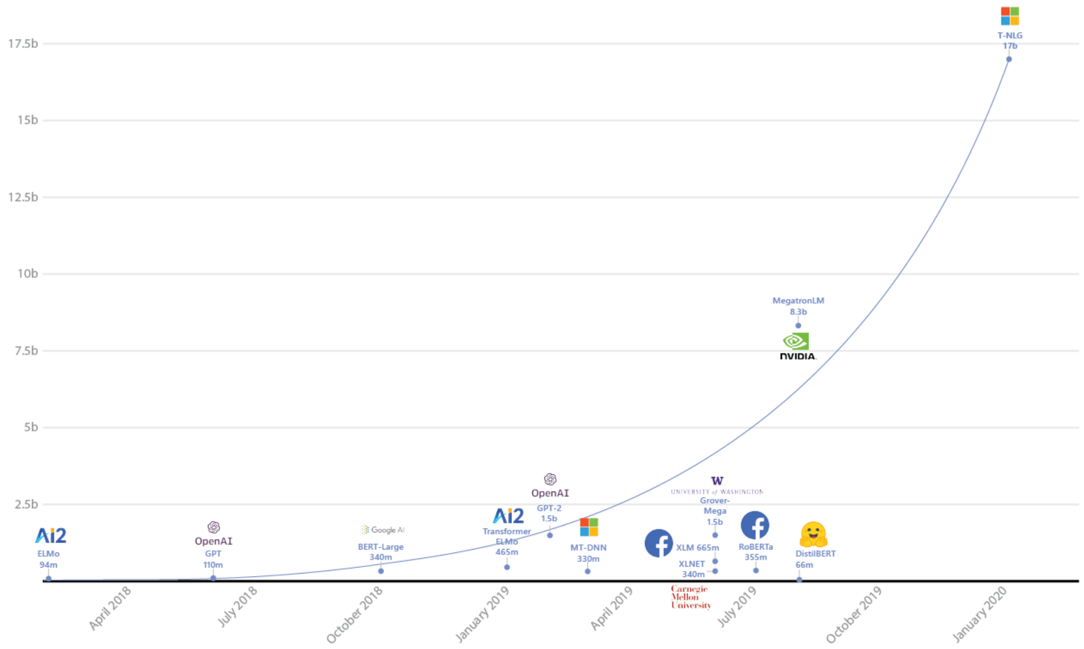
- Minimize NLL on parallel training set of N sentences [Hassan et al. 2018]:

$$\mathcal{L}_{\text{NLL}}(\theta) = -\sum_{n=1}^{N} \log p(\mathbf{y}^{(\mathbf{n})} | \mathbf{x}^{(\mathbf{n})}; \theta)$$

$$= -\sum_{n=1}^{N} \sum_{t=1}^{|\mathcal{T}|} \log p(\mathbf{y}_{\mathbf{t}}^{(\mathbf{n})} | \mathbf{y}_{<\mathbf{t}}^{(\mathbf{n})}, \mathbf{h}_{\mathbf{t}-\mathbf{1}}^{(\mathbf{n})}, \text{Att}(\text{Enc}(\mathbf{x}^{(\mathbf{n})}), \mathbf{y}_{<\mathbf{t}}^{(\mathbf{n})}, \mathbf{h}_{\mathbf{t}-\mathbf{1}}^{(\mathbf{n})}); \theta)$$

MOTIVATING DISTILLATION FOR NMT

The AI scaling law for LLMs



GOAL: Minimize neural machine translation model size while maintaining accuracy

OUTLINE

- Background & Motivation
- **Hinton-style Distillation**
  - ModelPT Distillation
- DistilBERT-style Distillation
- Sequence-Level Distillation
  - Greedy-search and beam-search sampling
- Hybrid Distillation
- Distillation in the Low Data Regime
- NMT Distillation Recommendations
- Future Work

KNOWLEDGE DISTILLATION [HINTON ET AL. 2015]

Negative log-likelihood for normal training

- Train small student network to learn from larger teacher network.
- Hinton et al. 2015 matches the student and teacher predictions via cross-entropy.
- For a classifier over $\mathcal{V}$ classes, minimize NLL (cross-entropy):

$$\mathcal{L}_{\mathrm{NLL}}(\theta) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{k=1}^{|\mathcal{V}|}\mathbb{1}\{y^{(n)} = k\}\log p(y^{(n)} = k|x;\theta)$$

between degenerate one-hot encoded data distribution (all mass in one class) and model distribution $p(y|x;\theta)$

KNOWLEDGE DISTILLATION [HINTON ET AL. 2015]

Small student network learns from large teacher network

- Train student classifier on the soft-labels of the teacher classifier rather than ground-truth labels
- Trained teacher classifier assigns probabilities to all labels
- "Relative probabilities of incorrect answers tell us a lot about how the [teacher] model tends to generalize"
- With learned teacher distribution $q(y|x;\theta_T)$, minimize cross entropy with teacher distribution on transfer set:

$$\mathcal{L}_{\text{KD}}(\theta;\theta_T) = -\frac{1}{N}\sum_{n=1}^{N} q(y^{(n)} = k|x^{(n)};\theta_T)\log p(y^{(n)} = k|x^{(n)};\theta)$$

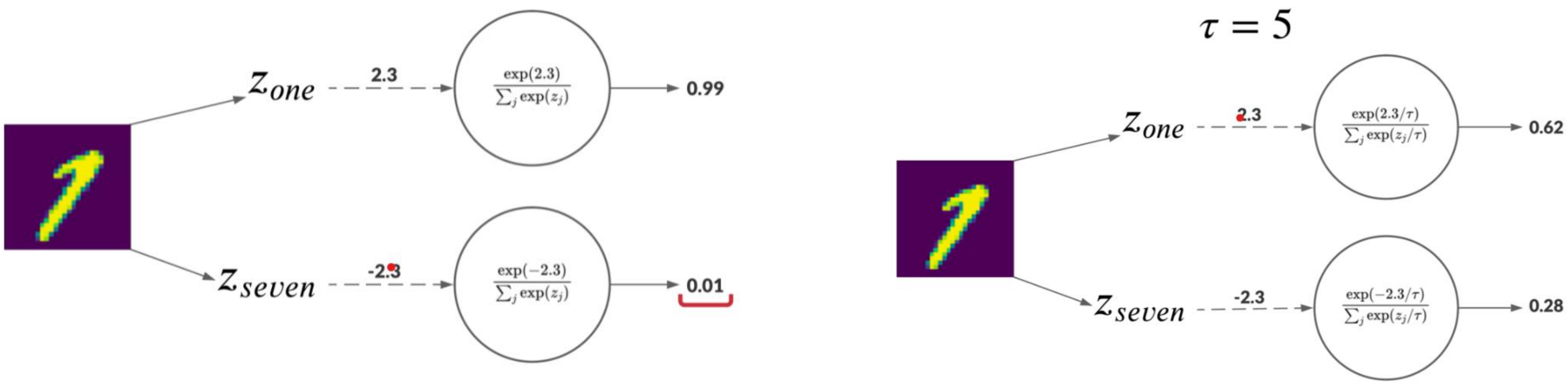
where $q_k := q(y = k|x;\theta_T) = \frac{\exp(z_k)}{\sum_{j=1}^{|\mathcal{V}|}\exp(z_j)}$ and $p_k := p(y = k|x;\theta) = \frac{\exp(w_k)}{\sum_{j=1}^{|\mathcal{V}|}\exp(w_j)}$

KNOWLEDGE DISTILLATION [HINTON ET AL. 2015]

Tempering distributions

- Compute the soft labels by using a tempered softmax (e.g. for student):

$$(p^\tau)_k = \frac{\exp(w_k/\tau)}{\sum_{j=1}^{|\mathcal{V}|} \exp(w_j/\tau)}$$



Taken from Distilling Knowledge in Neural Networks Blog [Sayak Paul]

KNOWLEDGE DISTILLATION [HINTON ET AL. 2015]

Interpolating objectives

- Interpolate between NLL and KD with mixing hyper-parameter α :

$$\mathcal{L}(\theta; \theta_T) = (1 - \alpha)\mathcal{L}_{\mathrm{NLL}}(\theta) + \alpha\mathcal{L}_{\mathrm{KD}}(\theta; \theta_T)$$

where $\mathcal{L}_{\mathrm{KD}} = \alpha\mathcal{L}_{CE}(\mathbf{q}^{\tau}, \mathbf{p}^{\tau}) + (\mathbf{1} - \alpha)\mathcal{L}_{\mathrm{CE}}(\mathbf{p}^{\tau}, \mathbf{y}_{\mathrm{true}})$

KNOWLEDGE DISTILLATION [HINTON ET AL. 2015]

Of gradients & weights

- If temperature is high compared to the logits:

$$\frac{\partial \mathcal{L}_{\text{CE}}}{\partial w_k} = \frac{1}{\tau}\left(\frac{\exp(w_k/\tau)}{\sum_{j=1}^{|\mathcal{V}|}\exp(w_k/\tau)} - \frac{\exp(z_k/\tau)}{\sum_{j=1}^{|\mathcal{V}|}\exp(z_k/\tau)}\right)$$

$$\approx \frac{1}{\tau}\left(\frac{1 + w_k/\tau}{N + \sum_{k=1}^{|\mathcal{V}|} w_k/\tau} - \frac{1 + z_k/\tau}{N + \sum_{k=1}^{|\mathcal{V}|} z_k/\tau}\right)$$

- If logits are zero-meaned for each example in transfer set:

$$\frac{\partial \mathcal{L}_{\text{CE}}}{\partial w_k} = \frac{1}{N\tau^2}(w_k - z_k)$$

KNOWLEDGE DISTILLATION FOR NMT

Word-level knowledge distillation

- With the tempered teacher distribution $q(y|x;\theta_T)$, minimize cross entropy with tempered student distribution:

$$\mathcal{L}_{\text{WORD-LEVEL}} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{|\mathcal{T}|}\sum_{k=1}^{|\mathcal{V}|} q(y_t^{(n)} = k|\mathbf{x}^{(\mathbf{n})}, \mathbf{y}_{<\mathbf{t}}^{(\mathbf{n})}) \log \mathbf{p}(\mathbf{y}_\mathbf{t}^{(\mathbf{n})} = \mathbf{k}|\mathbf{x}^{(\mathbf{n})}, \mathbf{y}_{<\mathbf{t}}^{(\mathbf{n})})$$

- Interpolate between NLL and Word-level KD with mixing hyper-parameter α:

$$\mathcal{L}(\theta;\theta_T) = (1-\alpha)\mathcal{L}_{\text{NLL}}(\theta) + \alpha\mathcal{L}_{\text{WORD-LEVEL}}(\theta;\theta_T)$$

KNOWLEDGE DISTILLATION FOR NMT

Detour on KL Divergence

- Measure of how far distribution p is from q:

$$D_{\mathrm{KL}}(p||q) := \mathbb{E}_{x\sim p}\left[\log \frac{p(x)}{q(x)}\right] = \sum_{x\in\mathcal{X}} p(x) \log \frac{p(x)}{q(x)}$$

- In terms of cross entropy:

$$D_{\mathrm{KL}}(p||q) = \sum_{x\in\mathcal{X}} p(x) \log p(x) - \sum_{x\in\mathcal{X}} p(x) \log q(x) = -H(p) + H(p,q)$$

where entropy is fixed for the training dataset.

KNOWLEDGE DISTILLATION [HINTON ET AL. 2015]

KL Divergence distillation objective

- Interpolate between NLL and KL divergence KDL:

$$\mathcal{L}(\theta; \theta_T) = (1 - \alpha)\mathcal{L}_{\text{NLL}}(\theta) + \alpha\mathcal{L}_{\text{KD}}(\theta; \theta_T)$$

where $\mathcal{L}_{KD} = D_{\text{KL}}(\mathbf{q}^{\tau} || \mathbf{p}^{\tau})$

TRAINING SETUP

Datasets and hyperparameters

- **Training set** - WMT21 DE->EN (same as transfer set in this part)
- **Validation sets** - WMT{13, 14, 18, 19, 20} DE->EN
- **Teacher architecture** - 24x4 encoder-decoder, attention_heads=16, hidden_size=1024, inner_sizer=4096
- **Student architectures**
  - **Slim architectures** - 1x1 and 3x3 encoder-decoder, attention_heads=4, hidden_size=256, inner_size=1024
  - **Wide architectures** - 1x1 and 3x4 encoder-decoder, attention_heads=16, hidden_size=1024, inner_size=4096
- **Temperature** – Perform grid search over T=[0.5, 1.0, 2.0, 5.0, 10.0]
- **Label smoothing** – Not recommended [Muller et al., 2019]
- **Optimizer** - Adam optimizer w/ inverse square root annealing schedule, lr=4e-4, warm_up_steps=1.5e4, steps=1.5e5

HINTON DISTILLATION FOR NMT

Teacher Results

WMT{13,14,18,19,20} German -> English

Wide 24x6	WMT13	WMT14	WMT18	WMT19	WMT20
sacreBLEU	45.4	47.1	37.1	34.4	40.1
Params.	468 M	468 M	468 M	468 M	468 M

HINTON DISTILLATION FOR NMT

1x1 Student Results

WMT20 German -> English

Slim 1x1	Temp=0.5	Temp=1.0	Temp=2.0	Temp=5.0	Temp=10.0	Params./Compression
Baseline	24.6	24.6	24.6	24.6	24.6	18.9 M/24.7
$\mathcal{L}_{\text{KD}} - \emptyset$	**25.3**	24.1	19.3	8.9	6.2	18.9 M/24.7
$\mathcal{L}_{\text{KD}} - \mathcal{L}_{\text{NLL}}$	**25.7**	24.5	22.2	12.5	10.4	18.9 M/24.7
Wide 1x1	**Temp=0.5**	**Temp=1.0**	**Temp=2.0**	**Temp=5.0**	**Temp=10.0**	**Params./Compression**
Baseline	32.7	32.7	32.7	32.7	32.7	95.0 M/5.0
$\mathcal{L}_{\text{KD}} - \emptyset$	32.4	**33.5**	31.4	27.3	14.8	95.0 M/5.0
$\mathcal{L}_{\text{KD}} - \mathcal{L}_{\text{NLL}}$	32.8	**34.6**	30.8	26.1	22.1	95.0 M/5.0
Weights	**1.0**	**1.0**	**2.0**	**40.0**	**200.0**	**-**

HINTON DISTILLATION FOR NMT

3x3 Student Results

WMT20 German -> English

Slim 3x3	Temp=0.5	Temp=1.0	Temp=2.0	Temp=5.0	Temp=10.0	Params./Compression
Baseline	29.6	29.6	29.6	29.6	29.6	21.9 M/21.3
$\mathcal{L}_{\text{KD}} - \emptyset$	**30.1**	**31.8**	29.3	24.1	17.3	21.9 M/21.3
$\mathcal{L}_{\text{KD}} - \mathcal{L}_{\text{NLL}}$	**31.3**	**32.5**	**31.9**	27.5	22.1	21.9 M/21.3
Wide 3x3	**Temp=0.5**	**Temp=1.0**	**Temp=2.0**	**Temp=5.0**	**Temp=10.0**	**Params./Compression**
Baseline	36.9	36.9	36.9	36.9	36.9	153.0 M/3.0
$\mathcal{L}_{\text{KD}} - \emptyset$	36.3	36.5	33.1	28.1	20.1	153.0 M/3.0
$\mathcal{L}_{\text{KD}} - \mathcal{L}_{\text{NLL}}$	**37.1**	**38.9**	34.4	26.7	23.5	153.0 M/3.0
Weights	**1.0**	**1.0**	**2.0**	**40.0**	**200.0**	**-**

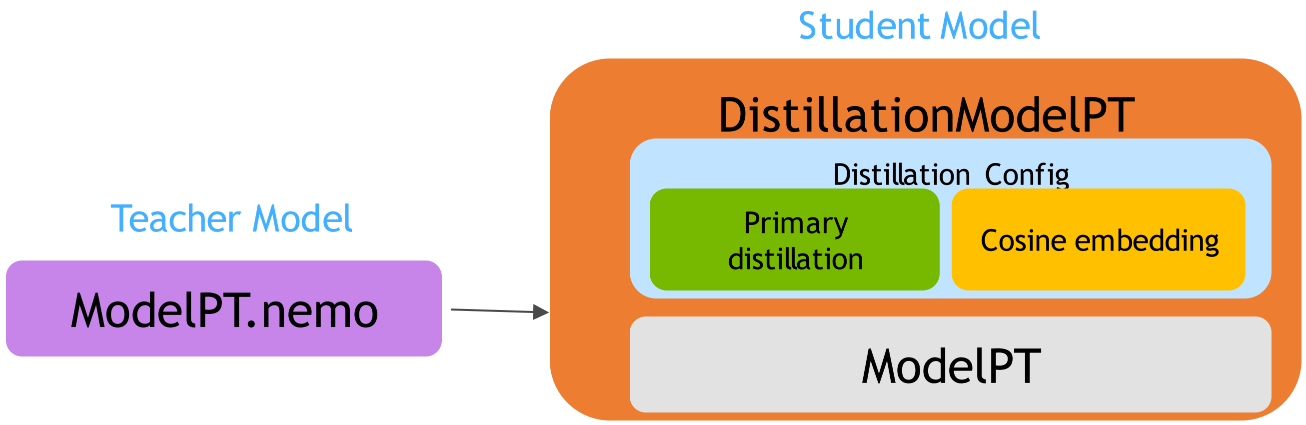
OUTLINE

- Background & Motivation
- Hinton-style Distillation
  - **ModelPT Distillation**
- DistilBERT-style Distillation
- Sequence-Level Distillation
  - Greedy-search and beam-search sampling
- Hybrid Distillation
- Distillation in the Low Data Regime
- NMT Distillation Recommendations
- Future Work

Disclaimer: PR in the works

HOW TO USE?

Nemo Guide for ModelPT Distillation [Som]




```python
from nemo.core.classes.distillation import DistillationModelPT

@hydra_runner(config_path="conf", config_name="aayn_base_distill")
def main(cfg: MTEncDecConfig) -> None:
    # training is managed by PyTorch Lightning
    trainer = Trainer(**cfg.trainer)

    # tokenizers will be trained and and tarred training data will be created if needed
    if cfg.model.preproc_out_dir is not None:
        MTDataPreproc(cfg=cfg.model, trainer=trainer)

    # experiment logs, checkpoints, and auto-resume are managed by exp_manager and PyTorch Lightning
    exp_manager(trainer, cfg.exp_manager)

    teacher_student_model = DistillationModelPT(cfg=cfg.model, trainer=trainer)

    if cfg.do_training:
        trainer.fit(teacher_student_model)
```

```python
@typecheck()
def forward(self, src, src_mask, tgt, tgt_mask):
    src_hiddens = self.encoder(input_ids=src, encoder_mask=src_mask)
    tgt_hiddens = self.decoder(
        input_ids=tgt, decoder_mask=tgt_mask, encoder_embeddings=src_hiddens, encoder_mask=src_mask
    )

    # Hinton distillation (either teacher/student)
    if self.is_being_distilled():
        self.log_softmax.log_softmax = False
        temperature = self.distill_cfg.get('temperature', 1.0)
        logits = self.log_softmax(hidden_states=tgt_hiddens)
        temp_logits = logits / temperature

        temp_log_probs = F.log_softmax(temp_logits, dim=-1)

        self.distillation_registration_step(log_prob=temp_log_probs)
        del temp_log_probs

        self.log_softmax.log_softmax = True

    log_probs = self.log_softmax(hidden_states=tgt_hiddens)

    return log_probs
```

OUTLINE

- Background & Motivation
- Hinton-style Distillation
  - ModelPT Distillation
- **DistilBERT-style Distillation**
- Sequence-Level Distillation
  - Greedy-search and beam-search sampling
- Hybrid Distillation
- Distillation in the Low Data Regime
- NMT Distillation Recommendations
- Future Work

DISTILBERT-STYLE DISTILLATION FOR NMT

Triple loss objective

- **Initialization:** instantiate student encoder-decoder by sampling 1 of every n layers from teacher encoder-decoder layers
    - E.g. 24x6 teacher->3x3 student: sample 1 every 8 from encoder & 1 every 2 from decoder
- **Triple loss linear combination:**

$$\mathcal{L} = \alpha_{\text{KD}}\mathcal{L}_{\text{KD}} + \alpha_{\text{NLL}}\mathcal{L}_{\text{NLL}} + \alpha_{\cos}\mathcal{L}_{\cos}$$

where $\mathcal{L}_{\cos}(\mathbf{h_s}, \mathbf{h_t}) = \mathbf{1} - \dfrac{\mathbf{h_s} \cdot \mathbf{h_t}}{||\mathbf{h_s}||||\mathbf{h_t}||}$

DISTILBERT-STYLE DISTILLATION FOR NMT

DistilBERT Ablation Study

Teacher: 32.7 sacreBLEU | Student: 1x1 Wide DE->EN, 95M params.

No Initialization	Temp=0.5	Temp=1.0	Temp=2.0	Temp=5.0	Temp=10.0	Max	max ΔS
$\emptyset - \mathcal{L}_{\text{NLL}} - \mathcal{L}_{\cos}$	-	32.3	-	-	-	32.3	-2.6
$\mathcal{L}_{KD} - \emptyset - \mathcal{L}_{\cos}$	**32.4**	32.1	29.0	24.1	18.2	32.4	-2.5
$\mathcal{L}_{KD} - \mathcal{L}_{\text{NLL}} - \emptyset$	32.8	**34.6**	30.8	26.1	22.1	34.6	-0.3
$\mathcal{L}_{KD} - \mathcal{L}_{\text{NLL}} - \mathcal{L}_{\cos}$	32.1	**34.9**	30.8	26.3	23.5	34.9	-
DistilBERT Initialization	**Temp=0.5**	**Temp=1.0**	**Temp=2.0**	**Temp=5.0**	**Temp=10.0**	**Max**	**max ΔS**
$\emptyset - \mathcal{L}_{\text{NLL}} - \mathcal{L}_{\cos}$	-	23.5	-	-	-	23.5	-0.9
$\mathcal{L}_{KD} - \emptyset - \mathcal{L}_{\cos}$	24.2	**23.7**	20.1	13.1	4.0	23.7	-0.7
$\mathcal{L}_{KD} - \mathcal{L}_{\text{NLL}} - \emptyset$	23.9	**24.3**	22.3	16.0	12.8	24.3	-0.1
$\mathcal{L}_{KD} - \mathcal{L}_{\text{NLL}} - \mathcal{L}_{\cos}$	**24.4**	24.2	22.2	16.0	11.2	24.4	-

OUTLINE

- Background & Motivation
- Hinton-style Distillation
  - ModelPT Distillation
- DistilBERT-style Distillation
- **Sequence-Level Distillation**
  - Greedy-search and beam-search sampling
  - Interpolation
- Hybrid Distillation
- Distillation in the Low Data Regime
- NMT Distillation Recommendations
- Future Work

SEQUENCE-LEVEL KNOWLEDGE DISTILLATION [KIM & RUSH, 2016]

Overview

- **Sequence level distribution:**

$$p(\mathbf{y}|\mathbf{x}) = \prod_{\mathbf{t}=\mathbf{1}}^{|\mathcal{T}|} \mathbf{p}(\mathbf{y_t}|\mathbf{x}, \mathbf{y_{<t}})$$

- Using the teacher sequence distribution $q(\mathrm{y}|\mathrm{x})$ over all possible sequences:

$$\mathcal{L}_{\text{SEQ-KD}} = -\sum_{\mathbf{y}\in\mathcal{T}} q(\mathbf{y}|\mathbf{x}) \log \mathbf{p}(\mathbf{y}|\mathbf{x})$$

- Due to the exponential number of terms, approximate with mode:

$$q(\mathbf{y}|\mathbf{x}) \sim \mathbb{1}\{\mathbf{y} = \text{argmax}_{\mathbf{y}\in\mathcal{T}} \mathbf{q}(\mathbf{y}|\mathbf{x})\}$$

SEQUENCE-LEVEL KNOWLEDGE DISTILLATION [KIM & RUSH, 2016]

Approximating the mode

- **Greedy-search sampling** - We can greedily sample the sequence of words
    - While cheap, from experiments, it is not as effective as beam search
- **Beam-search sampling (K=1)** - Run beam search with teacher model to obtain prediction $\hat{y}$ (expensive!)
- **Why?** - Large portion of teacher's distribution mass **q** lies in single output sequence
- Step 1: Train teacher model
- Step 2: Run beam search over training set with teacher to get "pseudo-label" dataset
- Step 3: Train the student network with cross entropy on new dataset

SEQUENCE-LEVEL KNOWLEDGE DISTILLATION [KIM & RUSH, 2016]

Sequence-level Interpolation

- Train student model as mixture of sequence level teacher-generated dataset and original training dataset:

$$\begin{aligned}\mathcal{L} &= -(1-\alpha)\mathcal{L}_{\text{SEQ-NLL}} + \alpha\mathcal{L}_{\text{SEQ-KD}} \\ &= -(1-\alpha)\log p(\mathbf{y}|\mathbf{x}) - \alpha\sum_{\mathbf{y}\in\mathcal{T}}\mathbf{q}(\mathbf{y}|\mathbf{x})\log \mathbf{p}(\mathbf{y}|\mathbf{x})\end{aligned}$$

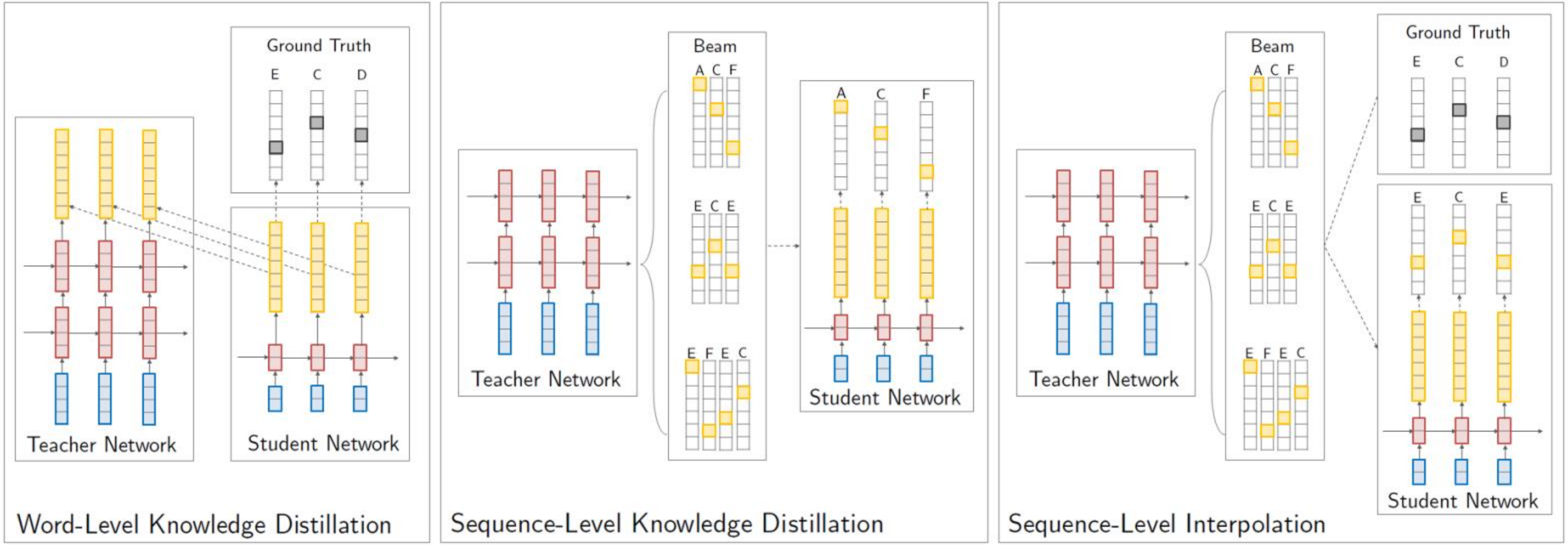
- Approximate second objective with beam search:

$$\begin{aligned}\mathcal{L}_{\text{SEQ-KD}} &\approx -\sum_{\mathbf{y}\in\mathcal{T}}\mathbb{1}\{\mathbf{y}=\hat{\mathbf{y}}\}\log \mathbf{p}(\mathbf{y}|\mathbf{x}) \\ &= -\log p(\mathbf{y}=\hat{\mathbf{y}}|\mathbf{x})\end{aligned}$$

- View interpolation as a form of regularization due to noisy data augmentation

SEQUENCE-LEVEL KNOWLEDGE DISTILLATION [KIM & RUSH, 2016]

Three variants



SEQUENCE-LEVEL KNOWLEDGE DISTILLATION [KIM & RUSH, 2016]

WMT20 German -> English

- **Temperature** - 1
- **1x1 wide student baseline** - 32.7 sacreBLEU
- **3x3 slim student baseline** - 29.6 sacreBLEU

1x1 wide student

Gold dataset mixing ratio	Teacher-generated dataset mixing ratio	sacreBLEU
0.1	0.9	33.1
0.34	0.66	32.4
0.66	0.34	33.2
0.9	0.1	32.9

3x3 slim student

Gold dataset mixing ratio	Teacher-generated dataset mixing ratio	sacreBLEU
0.1	0.9	30.6
0.34	0.66	30.4
0.66	0.34	30.3
0.9	0.1	29.4

OUTLINE

- Background & Motivation
- Hinton-style Distillation
  - ModelPT Distillation
- DistilBERT-style Distillation
- Sequence-Level Distillation
  - Greedy-search and beam-search sampling
- **Hybrid Distillation**
- Distillation in the Low Data Regime
- NMT Distillation Recommendations
- Future Work

HYBRID DISTILLATION

A hybrid of sequence-level interpolation and Hinton-style knowledge distillation

- Apply a hybrid of the two knowledge distillations:

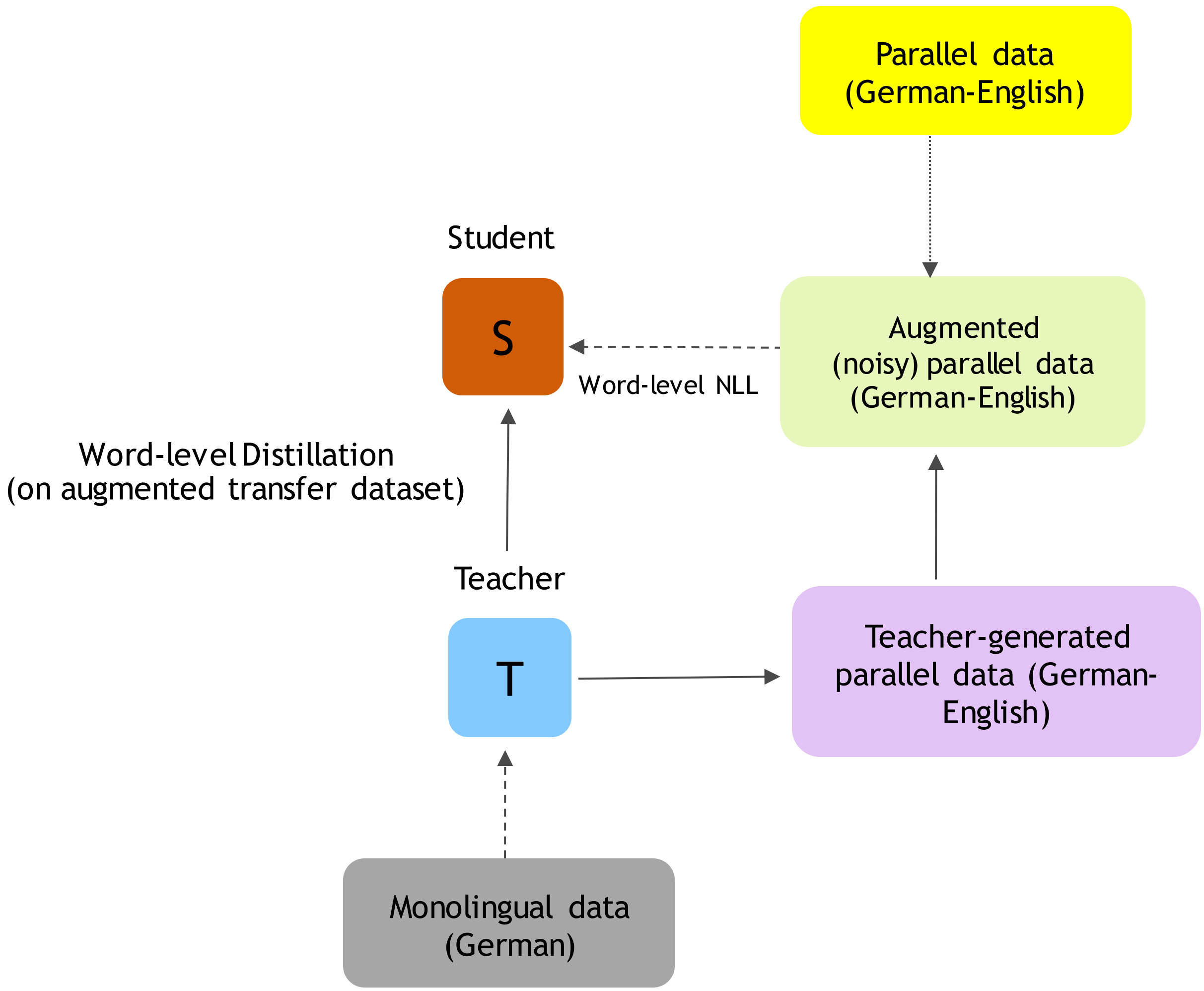
$$\begin{aligned}\mathcal{L} &= \alpha_{\text{SEQ-NLL}}\mathcal{L}_{\text{SEQ-NLL}} + \alpha_{\text{SEQ-KD}}\mathcal{L}_{\text{SEQ-KD}} + \alpha_{\text{WORD-KD}}\mathcal{L}_{\text{WORD-KD}} \\ &\approx -\alpha_{\text{SEQ-NLL}} \log \mathbf{p}(\mathbf{y}|\mathbf{x}) - \alpha_{\text{SEQ-KD}} \sum_{\mathbf{y}\in\mathcal{T}} \mathbf{q}(\mathbf{y}|\mathbf{x}) \log \mathbf{p}(\mathbf{y}|\mathbf{x}) + \alpha_{\text{WORD-KD}} \mathbf{D}_{\text{KL}}(\mathbf{q}||\mathbf{p})\end{aligned}$$

- Using the mode approximation:

$$\mathcal{L} = -\alpha_{\text{SEQ-NLL}} \log \mathbf{p}(\mathbf{y}|\mathbf{x}) - \alpha_{\text{SEQ-KD}} \log \mathbf{p}(\hat{\mathbf{x}}|\mathbf{x}) + \alpha_{\text{WORD-KD}} \mathbf{D}_{\text{KL}}(\mathbf{q}||\mathbf{p})$$

HYBRID DISTILLATION

Pipeline



HYBRID DISTILLATION

- Set-up: Use [0.34, 0.66] mixing probabilities and temperature of 1.0
- Same training & transfer datasets + new (noisy) teacher-generated dataset

German -> English

1x1 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	37.1	38.9	29.9	29.5	31.8	95M/5.0
$\mathcal{L}_{\mathrm{KD}} - \emptyset$	37.9	39.5	30.2	29.5	31.6	95M/5.0
$\mathcal{L}_{\mathrm{KD}} - \mathcal{L}_{\mathrm{NLL}}$	**38.5**	**40.9**	**31.5**	**30.7**	**33.6**	95M/5.0
3x3 slim	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	34.4	36.9	28.3	27.8	29.6	21.9M/21.3
$\mathcal{L}_{\mathrm{KD}} - \emptyset$	35.5	38.2	29.2	28.7	30.4	21.9M/21.3
$\mathcal{L}_{\mathrm{KD}} - \mathcal{L}_{\mathrm{NLL}}$	**35.6**	**38.2**	**29.3**	**28.8**	**30.4**	21.9M/21.3
3x3 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	34.4	36.9	28.2	27.8	29.6	153.0M/3.0
$\mathcal{L}_{\mathrm{KD}} - \emptyset$	42.9	44.8	35.3	33.2	37.9	153.0M/3.0
$\mathcal{L}_{\mathrm{KD}} - \mathcal{L}_{\mathrm{NLL}}$	**42.9**	**45.2**	**35.4**	**33.9**	**37.8**	153.0M/3.0
24x6 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	45.4	47.1	37.1	34.4	40.1	-

OUTLINE

- Background & Motivation
- Hinton-style Distillation
  - ModelPT Distillation
- DistilBERT-style Distillation
- Sequence-Level Distillation
  - Greedy-search and beam-search sampling
- Hybrid Distillation
- **Distillation in the Low Data Regime**
- NMT Distillation Recommendations
- Future Work

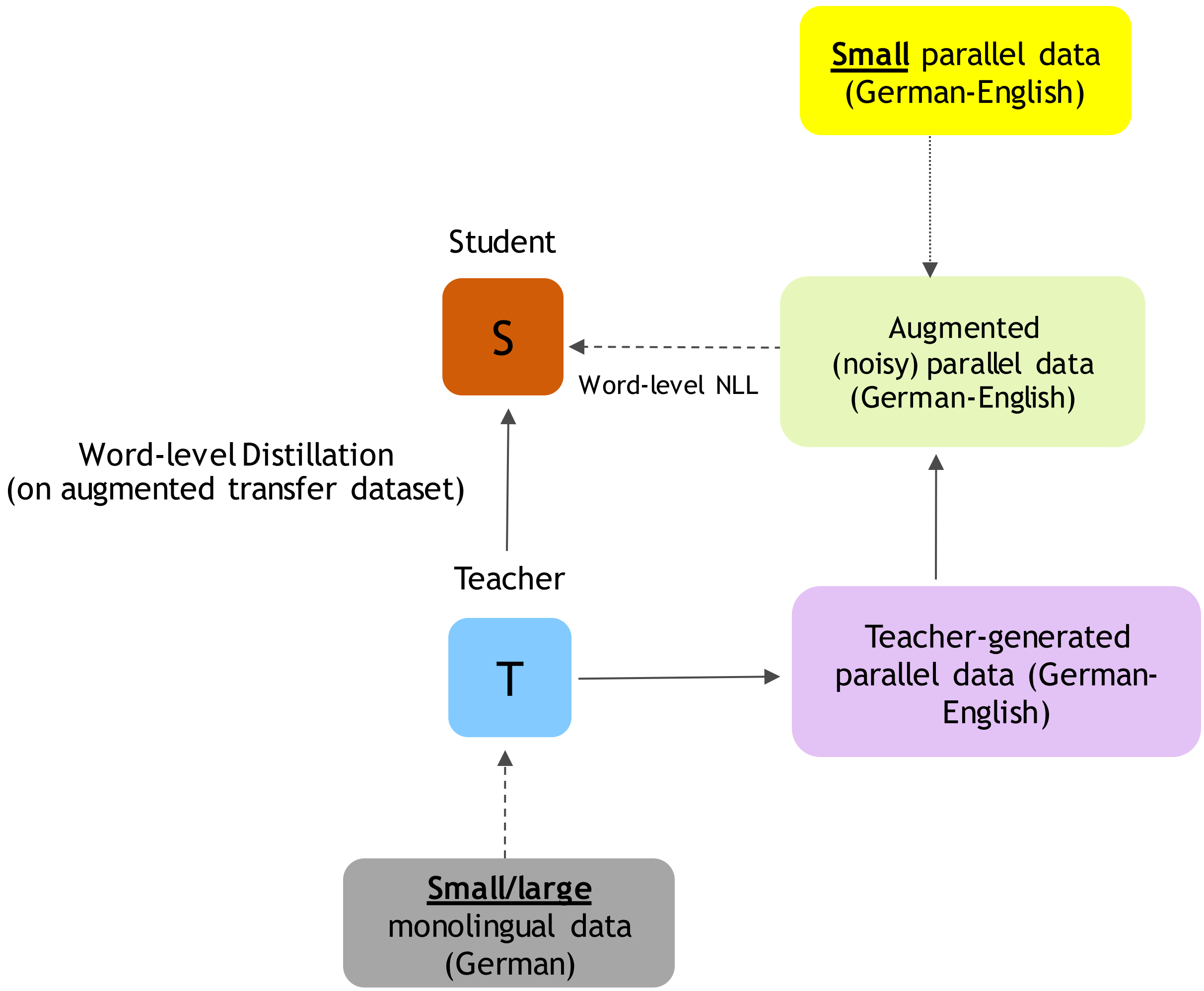
ENTER THE LOW DATA REGIME

Goal: training on only 5% of ground truth labels and pseudo-labels

- **Hinton et al. 2015** – "Soft targets allow student to generalize well from only 3% of the training set"
- **Approach** - Hybrid distillation with 5% of ground and pseudo labels (mixture 1:2) at temperature 1

HYBRID DISTILLATION

Low Data Regime



HYBRID DISTILLATION

Low Data Regime

1x1 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	37.1	38.9	29.9	29.5	31.8	95M/5.0
$\mathcal{L}_{\mathrm{KD}} - \emptyset$	37.1	39.4	29.8	29.2	31.3	95M/5.0
$\mathcal{L}_{\mathrm{KD}} - \mathcal{L}_{\mathrm{NLL}}$	**37.5**	**39.8**	**30.5**	**29.9**	**32.1**	95M/5.0
3x3 slim	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	34.4	36.9	28.3	27.8	29.6	21.9M/21.3
$\mathcal{L}_{\mathrm{KD}} - \emptyset$	35.7	37.8	28.8	28.4	30.87	21.9M/21.3
$\mathcal{L}_{\mathrm{KD}} - \mathcal{L}_{\mathrm{NLL}}$	**35.4**	**37.9**	**29.2**	**29.2**	**30.4**	21.9M/21.3
3x3 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	34.4	36.9	28.2	27.8	29.6	153.0M/3.0
$\mathcal{L}_{\mathrm{KD}} - \emptyset$	42.0	44.4	34.6	32.6	36.7	153.0M/3.0
$\mathcal{L}_{\mathrm{KD}} - \mathcal{L}_{\mathrm{NLL}}$	40.7	43.4	33.5	32.1	36	153.0M/3.0
24x6 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
Baseline	45.4	47.1	37.1	34.4	40.1	-

HYBRID DISTILLATION

Low Data Regime Relative Changes

1x1 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
$\mathcal{L}_{\text{KD}} - \emptyset$	-0.8	-0.1	-0.4	-0.3	-0.3	95M/5.0
$\mathcal{L}_{\text{KD}} - \mathcal{L}_{\text{NLL}}$	-1.0	-1.1	-1.0	-0.8	-1.5	95M/5.0
3x3 slim	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
$\mathcal{L}_{\text{KD}} - \emptyset$	+0.2	-0.4	-0.4	-0.3	+0.5	21.9M/21.3
$\mathcal{L}_{\text{KD}} - \mathcal{L}_{\text{NLL}}$	-0.2	-0.3	-0.1	-0.6	0.0	21.9M/21.3
3x3 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio
$\mathcal{L}_{\text{KD}} - \emptyset$	-0.9	-0.4	-0.7	-0.6	-1.2	153.0M/3.0
$\mathcal{L}_{\text{KD}} - \mathcal{L}_{\text{NLL}}$	-2.2	-1.8	-1.9	-1.8	-1.8	153.0M/3.0
3x3 wide	WMT13	WMT14	WMT18	WMT19	WMT20	Params./Compression ratio

OUTLINE

- Background & Motivation
- Hinton-style Distillation
  - ModelPT Distillation
- DistilBERT-style Distillation
- Sequence-Level Distillation
  - Greedy-search and beam-search sampling
- Hybrid Distillation
- Distillation in the Low Data Regime
- **NMT Distillation Recommendations**
- Future Work

NMT DISTILLATION RECOMMENDATIONS

How to proceed

- **How many parameters?** – In general, students with "reasonably-many" parameters ~90M (e.g. 1x1 wide) tend to exhibit desirable boosts; on the other hand, e.g. for 3x3 slim, with only ~20M, it's hard to learn from the teacher.

- **DistilBERT doesn't help** – With many ablation studies, it seems that initialization hurts performance for NMT & the DistilBERT setup significantly constrains the problem space.

- **Hinton [distillation] is not all you need -** If you want performance boosts >1-2 BLEU points, some form of semi-supervised distillation, sequence-level distillation, or interpolation is necessary (e.g. "hybrid distillation").

- **Use less data -** You can possibly get away with far less data than you might think. We used only 5% of all available labels + pseudo-labels and saw similar performance with hybrid distillation.

OUTLINE

- Background & Motivation
- Hinton-style Distillation
    - ModelPT Distillation
- DistilBERT-style Distillation
- Sequence-Level Distillation
    - Greedy-search and beam-search sampling
- Hybrid Distillation
- Distillation in the Low Data Regime
- NMT Distillation Recommendations
- **Future Work**

FUTURE WORK

On noisy students and unlabeled data

- We plan on running inference baselines to get a better idea of model efficiency.
- Semi-supervised distillation: with little labeled data and sizable unlabeled data.
- Use ideas from Self-training with Noisy Student [Xie et al., 2019] or Well-Read Students Learn Better [Chang et al., 2019] like pre-training students.

REFERENCES

- Distilling the Knowledge in a Neural Network [Hinton et al. 2015]
- Sequence-Level Knowledge Distillation [Kim et al., 2016]
- DistilBERT, a distilled version of BERT [Chaumond et al., 2019]
- Self-training with Noisy Student improves ImageNet Classification [Xie et al., 2019]
- Well-Read Students Learn Better [Chang et al., 2020]
- Achieving Human Parity on Automatic Chinese to English News Translation [Hassan et al., 2018]
- Big Self-Supervised Models are Strong Semi-Supervised Learners [Chen et al., 2020]
- When Does Label Smoothing Help? [Mueller et al., 2019]
- Softmax Tempering for Training Neural Machine Translation Models [Dabre & Fujita, 2020]

THANK YOU!

Questions?