Improving Group Fairness in Knowledge Distillation via Laplace Approximation of Early Exits

Edvin 24V0074 Sagar 24D0367

CS 769
Optimization in Machine Learning

2 May 2025

Overview

1. Recap from Seminar
2. Experiments And Results
3. Analysis And Future Work

Section Overview

1. Recap from Seminar
2. Experiments And Results
3. Analysis And Future Work

Recap from Seminar

- Knowledge Distillation as an effective way to distill knowledge from teacher to student

Recap from Seminar

- Knowledge Distillation as an effective way to distill knowledge from teacher to student
- Teacher Provides "Soft Targets"

Recap from Seminar

- Knowledge Distillation as an effective way to distill knowledge from teacher to student
- Teacher Provides "Soft Targets"
- Loss: kl divergence + cross-entropy

Recap from Seminar

- Knowledge Distillation as an effective way to distill knowledge from teacher to student
- Teacher Provides "Soft Targets"
- Loss: kl divergence + cross-entropy
- Student model relies on spurious correlations

Recap from Seminar

- Knowledge Distillation as an effective way to distill knowledge from teacher to student
- Teacher Provides "Soft Targets"
- Loss: kl divergence + cross-entropy
- Student model relies on spurious correlations
- Student's early Layers overconfident on hard instances

Recap from Seminar

- Knowledge Distillation as an effective way to distill knowledge from teacher to student
- Teacher Provides "Soft Targets"
- Loss: kl divergence + cross-entropy
- Student model relies on spurious correlations
- Student's early Layers overconfident on hard instances
- DEDIER loss

$$\mathcal{L}_{student} = \sum_{D_w} (1-\lambda) \cdot l_{ce} + \lambda \cdot \mathbf{wt} \cdot l_{kd}$$

where $\mathbf{wt} = \exp^{\beta.\mathbf{cm}.\alpha}$ and $\mathbf{cm}(\mathbf{p}) = \mathbf{p_{max}} - \max_{\mathbf{p_k} \in \mathbf{p} - \mathbf{p_{max}}} \mathbf{p_k}$

Recap from Seminar

- Two alternate approaches for estimating uncertainity in prediction in early exit layers.

Recap from Seminar

- Two alternate approaches for estimating uncertainity in prediction in early exit layers.
- [Meronen et al., 2023] used Laplace approximation for bayesian posterior at exit layer.

Recap from Seminar

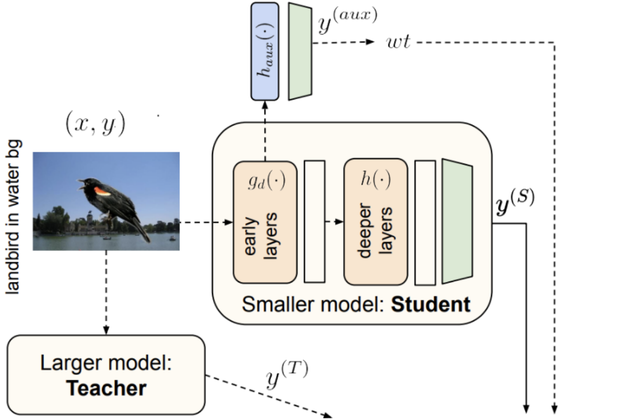
- Two alternate approaches for estimating uncertainity in prediction in early exit layers.
- [Meronen et al., 2023] used Laplace approximation for bayesian posterior at exit layer.
- [Jazbec et al., 2024] used AVCS based on Predictive-likelihood ratio to get confidence intervals for predictions.

Recap from Seminar

- Two alternate approaches for estimating uncertainity in prediction in early exit layers.
- [Meronen et al., 2023] used Laplace approximation for bayesian posterior at exit layer.
- [Jazbec et al., 2024] used AVCS based on Predictive-likelihood ratio to get confidence intervals for predictions.
- Experiment: Laplace Approximation based uncertainity estimate to reweight both the losses.

Recap from Seminar

Figure



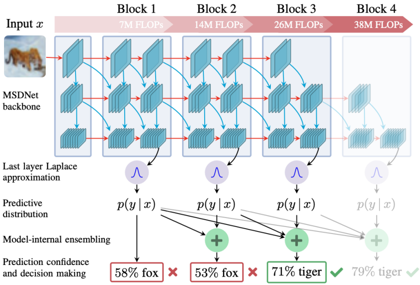
$$\mathcal{L}_{student} = (1-\lambda)\cdot l_{\mathrm{ce}}(y^{(S)}, y) + \lambda \cdot \boldsymbol{wt} \cdot l_{\mathrm{kd}}(y^{(S)}, y^{(T)})$$

Recap from Seminar

Figure

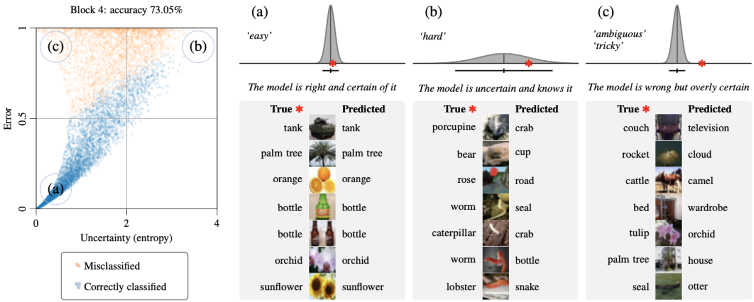
Recap from Seminar

Figure



Recap from Seminar

Figure



Recap from Seminar

- $$p(\boldsymbol{\theta} \mid \mathcal{D}_{\text{train}}) = \frac{p(\mathcal{D}_{\text{train}} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\mathcal{D}_{\text{train}}, \boldsymbol{\theta})\, d\boldsymbol{\theta}} = \frac{[\text{likelihood}] \times [\text{prior}]}{[\text{model evidence}]}$$

Recap from Seminar

- $$p(\boldsymbol{\theta} \mid \mathcal{D}_{\text{train}}) = \frac{p(\mathcal{D}_{\text{train}} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\mathcal{D}_{\text{train}}, \boldsymbol{\theta})\, d\boldsymbol{\theta}} = \frac{[\text{likelihood}] \times [\text{prior}]}{[\text{model evidence}]}$$
- MAP estimate can be found by maximising the unnormalised posterior:
$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \log p(\mathcal{D}_{\text{train}} \mid \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})$$

Recap from Seminar

- $$p(\boldsymbol{\theta} \mid \mathcal{D}_{\text{train}}) = \frac{p(\mathcal{D}_{\text{train}} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\mathcal{D}_{\text{train}}, \boldsymbol{\theta})\, d\boldsymbol{\theta}} = \frac{[\text{likelihood}] \times [\text{prior}]}{[\text{model evidence}]}$$
- MAP estimate can be found by maximising the unnormalised posterior:
$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \log p(\mathcal{D}_{\text{train}} \mid \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})$$
- Gaussian distribution via laplace approximation
$$p(\hat{\mathbf{z}}_i \mid \mathbf{x}_i) = \mathcal{N}(\hat{\mathbf{W}}_{\text{MAP}}^\top \hat{\phi}_i, (\hat{\phi}_i^\top \mathbf{V} \hat{\phi}_i)\mathbf{U})$$
$$\mathbf{V}^{-1} \otimes \mathbf{U}^{-1} = \mathbf{H}^{-1}$$

  where H is
$$\mathbf{H} := -\nabla^2_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}|\mathcal{D})\Big|_{\boldsymbol{\theta}_{\text{MAP}}}$$

Recap from Seminar

- $$p(\boldsymbol{\theta} \mid \mathcal{D}_{\text{train}}) = \frac{p(\mathcal{D}_{\text{train}} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(\mathcal{D}_{\text{train}}, \boldsymbol{\theta})\, d\boldsymbol{\theta}} = \frac{[\text{likelihood}] \times [\text{prior}]}{[\text{model evidence}]}$$
- MAP estimate can be found by maximising the unnormalised posterior:
$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \log p(\mathcal{D}_{\text{train}} \mid \boldsymbol{\theta}) + \log p(\boldsymbol{\theta})$$
- Gaussian distribution via laplace approximation
$$p(\hat{\mathbf{z}}_i \mid \mathbf{x}_i) = \mathcal{N}(\hat{\mathbf{W}}_{\text{MAP}}^{\top}\hat{\phi}_i, (\hat{\phi}_i^{\top}\mathbf{V}\hat{\phi}_i)\mathbf{U})$$
$$\mathbf{V}^{-1} \otimes \mathbf{U}^{-1} = \mathbf{H}^{-1}$$
where H is
$$\mathbf{H} := -\nabla^2_{\boldsymbol{\theta}} \log p(\boldsymbol{\theta}|\mathcal{D})\Big|_{\boldsymbol{\theta}_{\text{MAP}}}$$
- Samples
$$\hat{\mathbf{z}}_i^{(l)} = \hat{\mathbf{W}}_{\text{MAP}}^{\top}\hat{\phi}_i + (\hat{\phi}_i^{\top}\mathbf{V}\hat{\phi}_i)^{\frac{1}{2}}(\mathbf{L}\mathbf{g}^{(l)})$$
$\mathbf{g}^{(l)} \sim \mathcal{N}(0, \mathbf{I})$ and $\mathbf{L}$ is the Cholesky factor of $\mathbf{U}$

Section Overview

1. Recap from Seminar
2. Experiments And Results
3. Analysis And Future Work

Experiments And Results

- The MultiNLI dataset [Williams et al., 2018] was used.
- Determine if premise entails, contradicts, or is neutral given a hypothesis.
- Major Diffrences with DEDIER: The auxiliary reweighting is done in every epoch, as student trained for five epochs.
- The auxiliary network used is a simple, single layer network.
- Models Used
  - **Teacher model:** `bert-base-uncased` (12-layer BERT, hidden size 768)
  - **Student model:** `distilbert-base-uncased` (6-layer DistilBERT, hidden size 768)
  - **Auxiliary network:** One-layer linear classifier trained on the students third layer.

Experiments And Results

- Hyperparameters
  - **Teacher fine-tuning**
    - Epochs: $E_T = 3$
    - Learning rate: 2×10^{-5}
    - Optimizer: AdamW
  - **Student training**
    - Epochs: $E_S = 5$
    - Learning rate: 2×10^{-5}
    - Optimizer: AdamW
    - Distillation temperature: $\tau = 2.0$
- Training and Evaluation
  - Batch size: 16
  - Dataset: MultiNLI (via HuggingFace `datasets`)
  - Evaluation: Accuracy and per-group performance (negation vs. non-negation)
  - Tokenization: `bert-base-uncased` tokenizer with padding and truncation
  - Learning rate scheduler: Linear schedule with warm-up

Experiments And Results

Metric	Teacher	Student (Aux layer 3)	Student (Aux layer 6)	Group
Average Accuracy	0.845	0.835	0.832	All

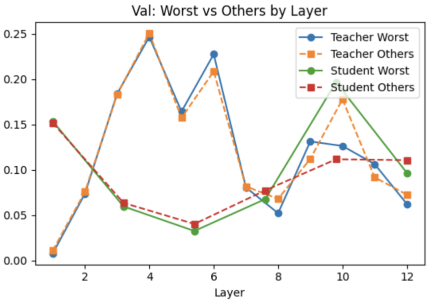
Table: Results of Dedier with Laplace

Experiments And Results

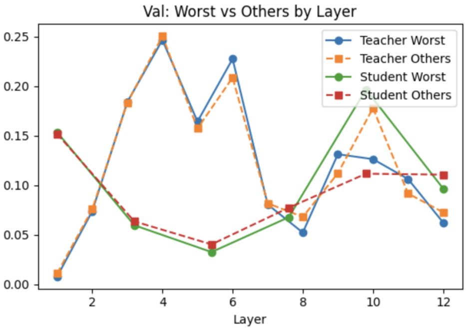
Metric	Teacher	Student
Final Test Accuracy	0.841	0.830

Table: Original DEDIER performance with same teacher

Experiments And Results



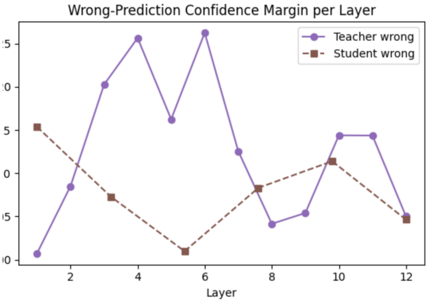
(a) Auxiliary network on layer 3



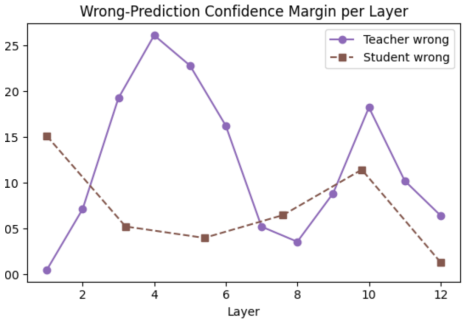
(b) Auxiliary network on layer 6 (last layer)

Figure: Confidence margins per layer on worst group predictions and all predictions

Experiments And Results



(a) Auxiliary network on layer 3



(b) Auxiliary network on layer 6 (last layer)

Figure: Confidence margins in student and teacher on the predictions that were wrong

Section Overview

Analysis And Future Work

- Minor improvement in accuracy of the student model by the proposed approach over DEDIER on MultiNLI
- Still more testing needed, on varied datasets and teacher models for justifying its use.
- The confidence margins are generally lower than one used in DEDIER work.
- Need to study effects of increasing layers of Aux network
- Hyperparameter tuning

Conclusion

- Cheap, effective uncertainty via Laplace in early exits.
- Uncertainty reweights KD loss in student models.
- Reduces reliance on simple features.

References

Jazbec, M., Forré, P., Mandt, S., Zhang, D., and Nalisnick, E. (2024).
Early-Exit Neural Networks with Nested Prediction Sets.
arXiv:2311.05931 [cs, stat].

Meronen, L., Trapp, M., Pilzer, A., Yang, L., and Solin, A. (2023).
Fixing Overconfidence in Dynamic Neural Networks.
Version Number: 4.

Williams, A., Nangia, N., and Bowman, S. R. (2018).
A broad-coverage challenge corpus for sentence understanding through inference.
In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.