Knowledge Distillation in Machine Learning

Samuel Cheng

University of Oklahoma

March 29, 2024

Introduction to Knowledge Distillation

- Definition: A process in machine learning that transfers knowledge from a large, complex model (teacher) to a smaller, simpler model (student).
- Purpose: Reduce computational resources, improve efficiency, and maintain accuracy.
- Applications: Deploying models on devices with limited hardware, improving inference speed, reducing energy consumption.

Introduction to Knowledge Distillation

- Definition: A process in machine learning that transfers knowledge from a large, complex model (teacher) to a smaller, simpler model (student).
- Purpose: Reduce computational resources, improve efficiency, and maintain accuracy.
- Applications: Deploying models on devices with limited hardware, improving inference speed, reducing energy consumption.

Introduction to Knowledge Distillation

- Definition: A process in machine learning that transfers knowledge from a large, complex model (teacher) to a smaller, simpler model (student).
- Purpose: Reduce computational resources, improve efficiency, and maintain accuracy.
- Applications: Deploying models on devices with limited hardware, improving inference speed, reducing energy consumption.

Knowledge Distillation Process

1. Train a larger, more complex model (teacher model).
2. Use the teacher model to generate soft labels on a dataset.
3. Train the smaller, simpler model (student model) using the soft labels.
4. The student model learns to mimic the teacher's behavior and generalization capabilities.

Knowledge Distillation Process

1. Train a larger, more complex model (teacher model).
2. Use the teacher model to generate soft labels on a dataset.
3. Train the smaller, simpler model (student model) using the soft labels.
4. The student model learns to mimic the teacher's behavior and generalization capabilities.

Knowledge Distillation Process

1. Train a larger, more complex model (teacher model).
2. Use the teacher model to generate soft labels on a dataset.
3. Train the smaller, simpler model (student model) using the soft labels.
4. The student model learns to mimic the teacher's behavior and generalization capabilities.

Knowledge Distillation Process

1. Train a larger, more complex model (teacher model).
2. Use the teacher model to generate soft labels on a dataset.
3. Train the smaller, simpler model (student model) using the soft labels.
4. The student model learns to mimic the teacher's behavior and generalization capabilities.

Soft Labels vs Hard Labels

- Hard Labels: Ground truth, discrete class labels (e.g., dog, cat, car).
- Soft Labels: Probability distributions over class labels generated by the teacher model.
- Benefits: Soft labels provide more information about the relationships between classes, making it easier for the student model to learn.

Soft Labels vs Hard Labels

- Hard Labels: Ground truth, discrete class labels (e.g., dog, cat, car).
- Soft Labels: Probability distributions over class labels generated by the teacher model.
- Benefits: Soft labels provide more information about the relationships between classes, making it easier for the student model to learn.

Soft Labels vs Hard Labels

- Hard Labels: Ground truth, discrete class labels (e.g., dog, cat, car).
- Soft Labels: Probability distributions over class labels generated by the teacher model.
- Benefits: Soft labels provide more information about the relationships between classes, making it easier for the student model to learn.

Temperature Scaling

Label distributions are often too "peak" as trained models are usually too confident. Only the ground truth label with probability close to 1

- Hinton et al 2015 introduces temperature to control the "smoothness" of the soft labels.

$$[\sigma(\mathrm{x};\mathrm{T})]_{\mathrm{i}} = \frac{\mathrm{e}^{\mathrm{T}\mathrm{x}_{\mathrm{i}}}}{\sum_{\mathrm{j}} \mathrm{e}^{\mathrm{T}\mathrm{x}_{\mathrm{j}}}}$$

  - Higher temperatures: Produce smoother probability distributions, making it easier for the student model to learn.
  - Lower temperatures: Produce more peaky probability distributions, closer to hard labels.
- Hyperparameter tuning is necessary to find the optimal temperature.

Temperature Scaling

Label distributions are often too “peak” as trained models are usually too confident. Only the ground truth label with probability close to 1

- Hinton et al 2015 introduces temperature to control the “smoothness” of the soft labels.

$$[\sigma(x;T)]_i = \frac{e^{Tx_i}}{\sum_j e^{Tx_j}}$$

  - Higher temperatures: Produce smoother probability distributions, making it easier for the student model to learn.
  - Lower temperatures: Produce more peaky probability distributions, closer to hard labels.
- Hyperparameter tuning is necessary to find the optimal temperature.

Temperature Scaling

Label distributions are often too "peak" as trained models are usually too confident. Only the ground truth label with probability close to 1

- Hinton et al 2015 introduces temperature to control the "smoothness" of the soft labels.

$$[\sigma(\mathrm{x};\mathrm{T})]_\mathrm{i} = \frac{\mathrm{e}^{\mathrm{T}\mathrm{x}_\mathrm{i}}}{\sum_\mathrm{j} \mathrm{e}^{\mathrm{T}\mathrm{x}_\mathrm{j}}}$$

  - Higher temperatures: Produce smoother probability distributions, making it easier for the student model to learn.
  - Lower temperatures: Produce more peaky probability distributions, closer to hard labels.
- Hyperparameter tuning is necessary to find the optimal temperature.

Temperature Scaling

Label distributions are often too "peak" as trained models are usually too confident. Only the ground truth label with probability close to 1

- Hinton et al 2015 introduces temperature to control the "smoothness" of the soft labels.

$$[\sigma(\mathrm{x}; \mathrm{T})]_{\mathrm{i}} = \frac{\mathrm{e}^{\mathrm{T}\mathrm{x}_{\mathrm{i}}}}{\sum_{\mathrm{j}} \mathrm{e}^{\mathrm{T}\mathrm{x}_{\mathrm{j}}}}$$

  - Higher temperatures: Produce smoother probability distributions, making it easier for the student model to learn.
  - Lower temperatures: Produce more peaky probability distributions, closer to hard labels.
- Hyperparameter tuning is necessary to find the optimal temperature.

Loss Functions for Knowledge Distillation

- Combine two types of loss functions:
  1. Student loss: Cross-entropy loss between the ground-truth and student's predictions
  2. Distillation loss: Cross-entropy loss between soft labels (teacher's predictions) and student's prediction with temperature > 1

$$\mathrm{L}(\mathrm{x};\mathrm{W}) = \alpha \underbrace{\mathrm{H}(\mathrm{y}, \sigma(\mathrm{z_s}; \mathrm{T}=1))}_{\text{student loss}} + \beta \underbrace{\mathrm{H}(\sigma(\mathrm{z_t}; \mathrm{T}=\tau), \sigma(\mathrm{z_s}, \mathrm{T}=\tau))}_{\text{distillation loss}}$$

Loss Functions for Knowledge Distillation

- Combine two types of loss functions:
  1. Student loss: Cross-entropy loss between the ground-truth and student's predictions
  2. Distillation loss: Cross-entropy loss between soft labels (teacher's predictions) and student's prediction with temperature > 1

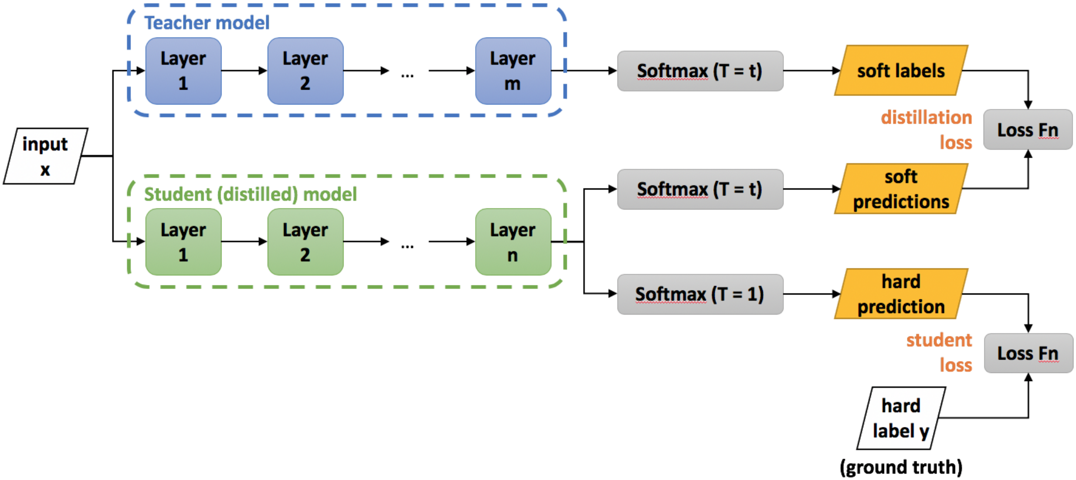
$$L(x; W) = \alpha \underbrace{H(y, \sigma(z_s; T = 1))}_{\text{student loss}} + \beta \underbrace{H(\sigma(z_t; T = \tau), \sigma(z_s, T = \tau))}_{\text{distillation loss}}$$

Loss Functions for Knowledge Distillation

- Combine two types of loss functions:
  1. Student loss: Cross-entropy loss between the ground-truth and student's predictions
  2. Distillation loss: Cross-entropy loss between soft labels (teacher's predictions) and student's prediction with temperature > 1

$$L(x; W) = \alpha \underbrace{H(y, \sigma(z_s; T = 1))}_{\text{student loss}} + \beta \underbrace{H(\sigma(z_t; T = \tau), \sigma(z_s, T = \tau))}_{\text{distillation loss}}$$

Knowledge distillation (Hinton et al 2015)



Result

System	Test Frame Accuracy	WER
Baseline	58.9%	10.9%
10xEnsemble	61.1%	10.7%
Distilled Single model	60.8%	10.7%

Table 1: Frame classification accuracy and WER showing that the distilled single model performs about as well as the averaged predictions of 10 models that were used to create the soft targets.

System & training set	Train Frame Accuracy	Test Frame Accuracy
Baseline (100% of training set)	63.4%	58.9%
Baseline (3% of training set)	67.3%	44.5%
Soft Targets (3% of training set)	65.4%	57.0%

Table 5: Soft targets allow a new model to generalize well from only 3% of the training set. The soft targets are obtained by training on the full training set.

Advantages of Distillation

- Faster inference times for the student model.
- Reduced memory and computational requirements.
- Maintains accuracy close to the teacher model.
- Enables deployment on devices with limited hardware resources.
- Generalize well even with a fraction of training data

Challenges and Future Directions

- Choosing appropriate teacher and student model architectures.
- Determining the optimal temperature scaling value.
- Exploring the potential of distilling multiple teacher models.
- Investigating new loss functions and distillation techniques.

Links

- Neural network distiller explanation
- Hinton's presentation and video