Knowledge Distillation

Intelligent Architectures (5LIL0)

dr Alexios Balatsoukas-Stimming

Department of Electrical Engineering, Electronic Systems Group

EINDHOVEN UNIVERSITY OF TECHNOLOGY

Knowledge Distillation

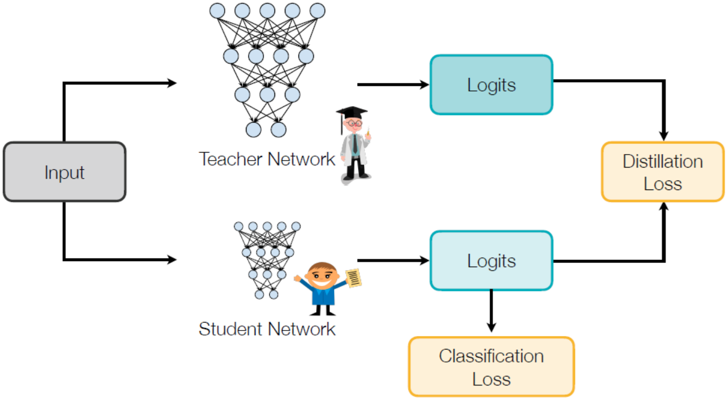
Definition: Knowledge Distillation (KD)

The transfer of knowledge from a large **teacher** network to a smaller **student** network.

Knowledge Distillation

Definition: Knowledge Distillation (KD)

The transfer of knowledge from a large **teacher** network to a smaller **student** network.



Prof. Song Han, "Knowledge Distillation" MIT 6.5940

Temperature

- Recall the softmax operation that produces output probabilities y_i:

$$y_i = \mathrm{softmax}(\mathbf{z}) = \frac{e^{z_i}}{\sum_{k=1}^{m} e^{z_k}}$$

- The outputs of the NN z_i are called **logits**

Temperature

- Recall the softmax operation that produces output probabilities y_i:

$$y_i = \text{softmax}(\mathbf{z}) = \frac{e^{z_i}}{\sum_{k=1}^{m} e^{z_k}}$$

- The outputs of the NN z_i are called **logits**

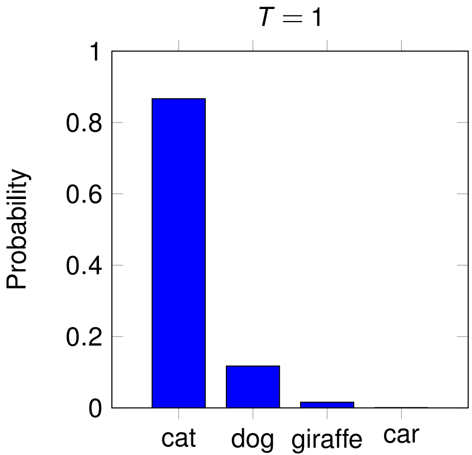
Definition: Softmax with temperature

$$y_i = \text{softmax}(\mathbf{z}, T) = \frac{e^{\frac{z_i}{T}}}{\sum_{k=1}^{m} e^{\frac{z_k}{T}}},$$

where T is the temperature parameter.

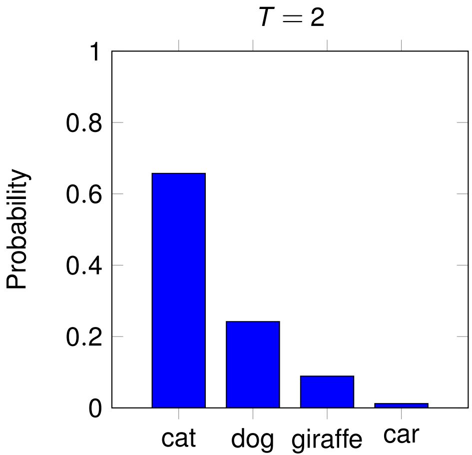
Why Temperature?

- Increasing T makes the output distribution more uniform



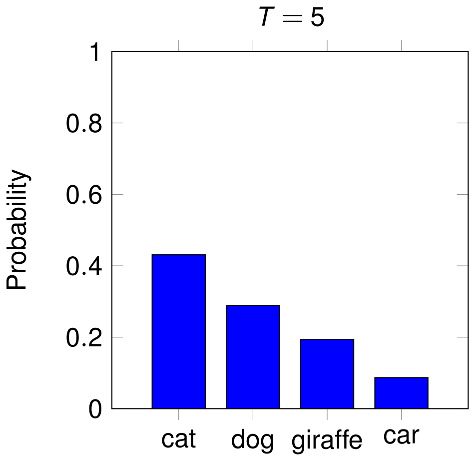
Why Temperature?

- Increasing T makes the output distribution more uniform



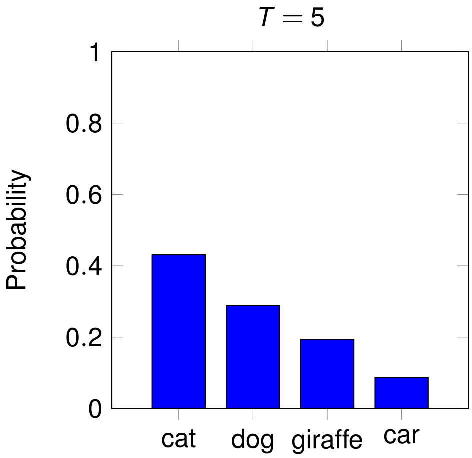
Why Temperature?

- Increasing T makes the output distribution more uniform



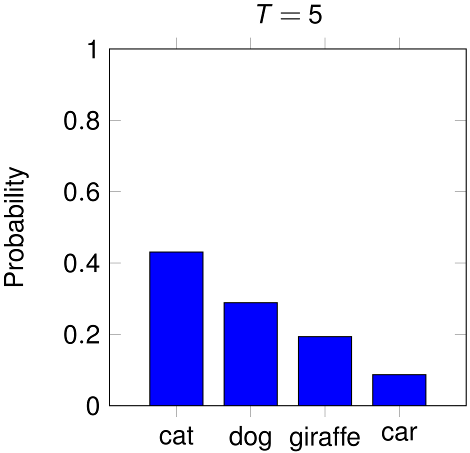
Why Temperature?

- Increasing T makes the output distribution more uniform
- Soft labels add more constraints on the parameters of the student network



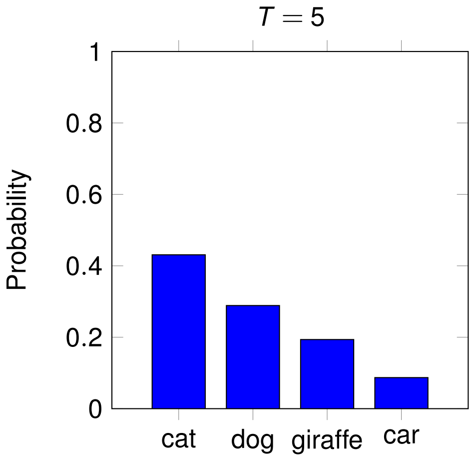
Why Temperature?

- Increasing T makes the output distribution more uniform
- Soft labels add more constraints on the parameters of the student network
- A form of **regularization**!



Why Temperature?

- Increasing T makes the output distribution more uniform
- Soft labels add more constraints on the parameters of the student network
- A form of **regularization**!
- "Distillation" is refinement at high temperature (e.g., alcohol)

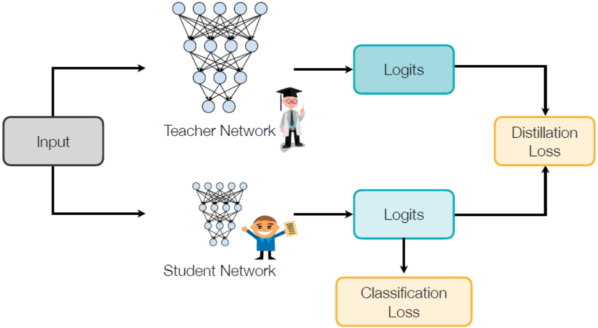


TU/e

Student Training (1/2)

Let's define:

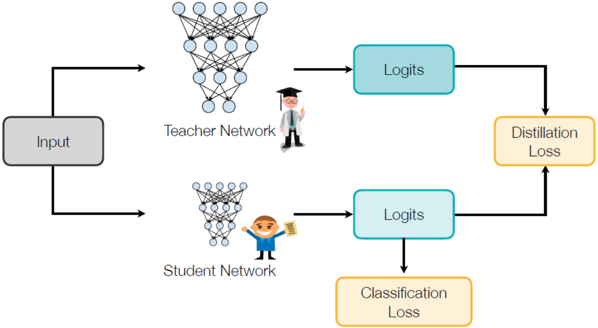
- Teacher output with $T > 1$: $\mathbf{y}_t$
- Student output with $T > 1$: $\mathbf{y}_s$
- Student output with $T = 1$: $\hat{\mathbf{y}}_s$
- Dataset label: $\mathbf{y}$



Student Training (1/2)

Let's define:

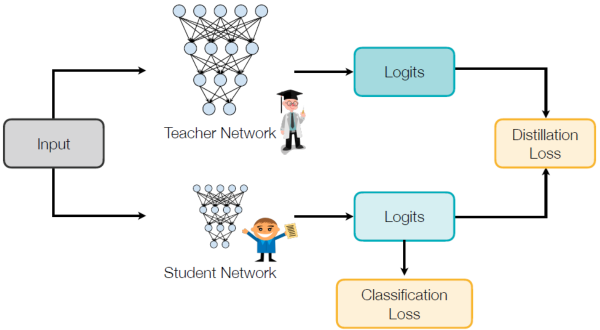
- Teacher output with $T > 1$: $\mathbf{y}_t$
- Student output with $T > 1$: $\mathbf{y}_s$
- Student output with $T = 1$: $\hat{\mathbf{y}}_s$
- Dataset label: $\mathbf{y}$



1. **Distillation loss:**
   - Kullback-Leibler divergence loss: $L_{\text{KL}}(\mathbf{y}_t, \mathbf{y}_s) = \mathbf{y}_t^\top \log\left(\frac{\mathbf{y}_t}{\mathbf{y}_s}\right)$

G. Hinton, O. Vinyals, J. Dean, "Distilling the Knowledge in a Neural Network," 2015.

Student Training (1/2)



Let's define:

- Teacher output with $T > 1$: $\mathbf{y}_t$
- Student output with $T > 1$: $\mathbf{y}_s$
- Student output with $T = 1$: $\hat{\mathbf{y}}_s$
- Dataset label: $\mathbf{y}$

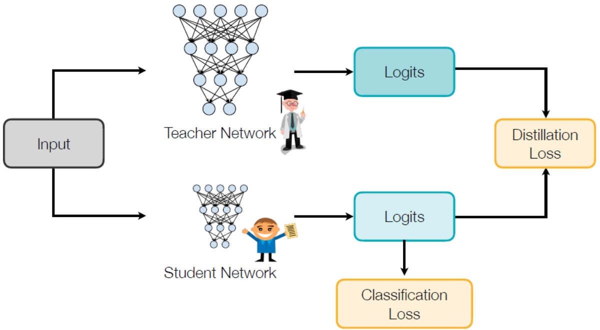
1. **Distillation loss:**
   - Kullback-Leibler divergence loss: $L_{\text{KL}}(\mathbf{y}_t, \mathbf{y}_s) = \mathbf{y}_t^\top \log\left(\frac{\mathbf{y}_t}{\mathbf{y}_s}\right)$
   - Cross-entropy loss: $L_{\text{CE}}(\mathbf{y}_t, \mathbf{y}_s) = -\mathbf{y}_t^\top \log(\mathbf{y}_s)$

G. Hinton, O. Vinyals, J. Dean, "Distilling the Knowledge in a Neural Network," 2015.

Student Training (1/2)

Let's define:

- Teacher output with $T > 1$: $\mathbf{y}_t$
- Student output with $T > 1$: $\mathbf{y}_s$
- Student output with $T = 1$: $\hat{\mathbf{y}}_s$
- Dataset label: $\mathbf{y}$



1. **Distillation loss:**
   - Kullback-Leibler divergence loss: $L_{\text{KL}}(\mathbf{y}_t, \mathbf{y}_s) = \mathbf{y}_t^\top \log\left(\frac{\mathbf{y}_t}{\mathbf{y}_s}\right)$
   - Cross-entropy loss: $L_{\text{CE}}(\mathbf{y}_t, \mathbf{y}_s) = -\mathbf{y}_t^\top \log(\mathbf{y}_s)$
2. **Classification loss:**
   - Cross-entropy loss: $L_{\text{CE}}(\mathbf{y}, \hat{\mathbf{y}}_s) = -\mathbf{y}^\top \log(\hat{\mathbf{y}}_s)$

G. Hinton, O. Vinyals, J. Dean, "Distilling the Knowledge in a Neural Network," 2015.

Kullback-Leibler Divergence Loss vs Cross-Entropy Loss

- General relationship between KL and CE:

$$L_{\mathrm{KL}}(\mathbf{y}, \hat{\mathbf{y}}) = \mathbf{y}^{\top} \log \left(\frac{\mathbf{y}}{\hat{\mathbf{y}}} \right)$$

Kullback-Leibler Divergence Loss vs Cross-Entropy Loss

- General relationship between KL and CE:

$$L_{\mathrm{KL}}(\mathbf{y}, \hat{\mathbf{y}}) = \mathbf{y}^\top \log\left(\frac{\mathbf{y}}{\hat{\mathbf{y}}}\right) = \mathbf{y}^\top \log \mathbf{y} - \mathbf{y}^\top \log \hat{\mathbf{y}}$$

Kullback-Leibler Divergence Loss vs Cross-Entropy Loss

- General relationship between KL and CE:

$$L_{\mathrm{KL}}(\mathbf{y},\hat{\mathbf{y}}) = \mathbf{y}^\top \log\left(\frac{\mathbf{y}}{\hat{\mathbf{y}}}\right) = \mathbf{y}^\top \log \mathbf{y} - \mathbf{y}^\top \log \hat{\mathbf{y}} = L_{\mathrm{CE}}(\mathbf{y},\hat{\mathbf{y}}) - H(\mathbf{y})$$

Kullback-Leibler Divergence Loss vs Cross-Entropy Loss

- General relationship between KL and CE:

$$L_{\mathrm{KL}}(\mathbf{y}, \hat{\mathbf{y}}) = \mathbf{y}^\top \log\left(\frac{\mathbf{y}}{\hat{\mathbf{y}}}\right) = \mathbf{y}^\top \log \mathbf{y} - \mathbf{y}^\top \log \hat{\mathbf{y}} = L_{\mathrm{CE}}(\mathbf{y}, \hat{\mathbf{y}}) - H(\mathbf{y})$$

- When $\mathbf{y}$ is one-hot encoded, $H(\mathbf{y}) = 0$, so $L_{\mathrm{KL}}(\mathbf{y}, \hat{\mathbf{y}}) = L_{\mathrm{CE}}(\mathbf{y}, \hat{\mathbf{y}})$

Kullback-Leibler Divergence Loss vs Cross-Entropy Loss

- General relationship between KL and CE:

$$L_{\text{KL}}(\mathbf{y}, \hat{\mathbf{y}}) = \mathbf{y}^\top \log\left(\frac{\mathbf{y}}{\hat{\mathbf{y}}}\right) = \mathbf{y}^\top \log \mathbf{y} - \mathbf{y}^\top \log \hat{\mathbf{y}} = L_{\text{CE}}(\mathbf{y}, \hat{\mathbf{y}}) - H(\mathbf{y})$$

- When $\mathbf{y}$ is one-hot encoded, $H(\mathbf{y}) = 0$, so $L_{\text{KL}}(\mathbf{y}, \hat{\mathbf{y}}) = L_{\text{CE}}(\mathbf{y}, \hat{\mathbf{y}})$
- In knowledge distillation, $\mathbf{y}$ are soft labels and $L_{\text{KL}}(\mathbf{y}, \hat{\mathbf{y}}) \neq L_{\text{CE}}(\mathbf{y}, \hat{\mathbf{y}})$

Kullback-Leibler Divergence Loss vs Cross-Entropy Loss

- General relationship between KL and CE:

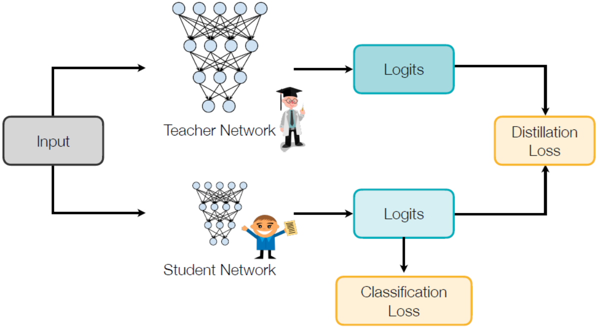
$$L_{\mathrm{KL}}(\mathbf{y},\hat{\mathbf{y}}) = \mathbf{y}^\top \log\left(\frac{\mathbf{y}}{\hat{\mathbf{y}}}\right) = \mathbf{y}^\top \log \mathbf{y} - \mathbf{y}^\top \log \hat{\mathbf{y}} = L_{\mathrm{CE}}(\mathbf{y},\hat{\mathbf{y}}) - H(\mathbf{y})$$

- When **y** is one-hot encoded, $H(\mathbf{y}) = 0$, so $L_{\mathrm{KL}}(\mathbf{y},\hat{\mathbf{y}}) = L_{\mathrm{CE}}(\mathbf{y},\hat{\mathbf{y}})$
- In knowledge distillation, **y** are soft labels and $L_{\mathrm{KL}}(\mathbf{y},\hat{\mathbf{y}}) \neq L_{\mathrm{CE}}(\mathbf{y},\hat{\mathbf{y}})$
- **However:** $\frac{\partial H(\mathbf{y})}{\partial \hat{\mathbf{y}}} = 0$, so $L_{\mathrm{CE}}(\mathbf{y},\hat{\mathbf{y}})$ and $L_{\mathrm{KL}}(\mathbf{y},\hat{\mathbf{y}})$ are interchangeable for training!

Student Training (2/2)

Knowledge distillation:

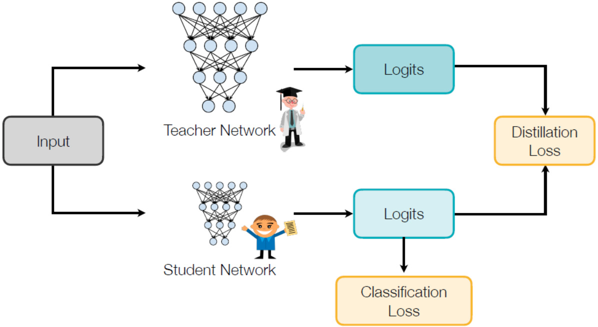
1. Pre-train teacher
2. Fix teacher weights
3. Train student



Student Training (2/2)

Knowledge distillation:

1. Pre-train teacher
2. Fix teacher weights
3. Train student



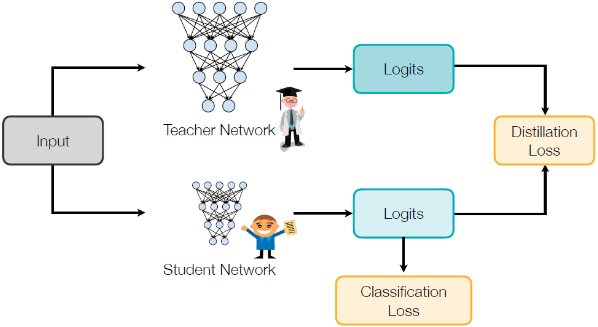
- **Student loss function:**

$$L(\mathbf{y}, \hat{\mathbf{y}}_s, \mathbf{y}_s, \mathbf{y}_t) = \alpha L_{\mathrm{CE}}(\mathbf{y}, \hat{\mathbf{y}}_s) + (1-\alpha) T^2 L_{\mathrm{CE}}(\mathbf{y}_t, \mathbf{y}_s), \quad \alpha \in [0, 1]$$

Student Training (2/2)

Knowledge distillation:

1. Pre-train teacher
2. Fix teacher weights
3. Train student



- **Student loss function:**

$$L(\mathbf{y}, \hat{\mathbf{y}}_s, \mathbf{y}_s, \mathbf{y}_t) = \alpha L_{\text{CE}}(\mathbf{y}, \hat{\mathbf{y}}_s) + (1-\alpha) T^2 L_{\text{CE}}(\mathbf{y}_t, \mathbf{y}_s), \quad \alpha \in [0, 1]$$

- Due to the use of softmax$(\mathbf{z}, T)$, $L_{\text{CE}}(\mathbf{y}_t, \mathbf{y}_s)$ is T^2 times smaller than $L_{\text{CE}}(\mathbf{y}, \hat{\mathbf{y}}_s)$

G. Hinton, O. Vinyals, J. Dean, "Distilling the Knowledge in a Neural Network," 2015.

A Surprising Result

- **Experiment:** Remove all instances of “3” from the training set of the student

[1] G. Hinton, O. Vinyals, J. Dean, “Distilling the Knowledge in a Neural Network,” 2015.

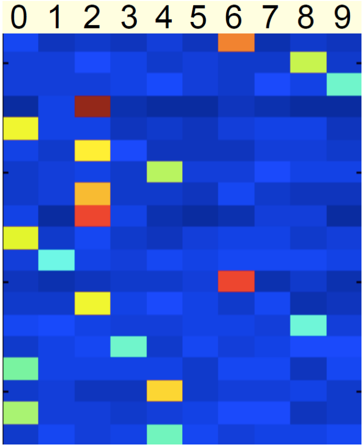
A Surprising Result

- **Experiment:** Remove all instances of “3” from the training set of the student
- **Result:** 87% of threes in the test set are classified correctly [1]!

[1] G. Hinton, O. Vinyals, J. Dean, “Distilling the Knowledge in a Neural Network,” 2015.

A Surprising Result

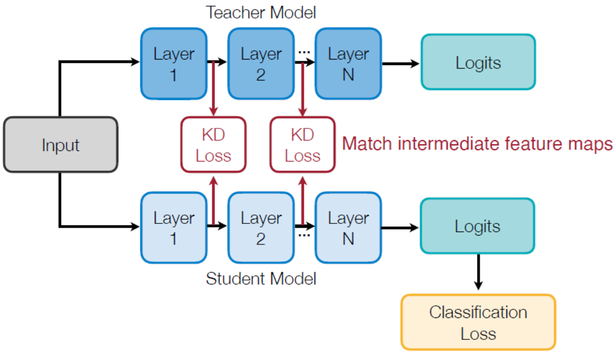
- **Experiment:** Remove all instances of “3” from the training set of the student
- **Result:** 87% of threes in the test set are classified correctly [1]!
- **How can this be?** The teacher’s “**dark knowledge**” [2] becomes accessible!



[1] G. Hinton, O. Vinyals, J. Dean, “Distilling the Knowledge in a Neural Network,” 2015.
[2] G. Hinton, O. Vinyals, J. Dean, “Dark knowledge,” TTIC Distinguished Lecture Series, 2014.

Matching Features

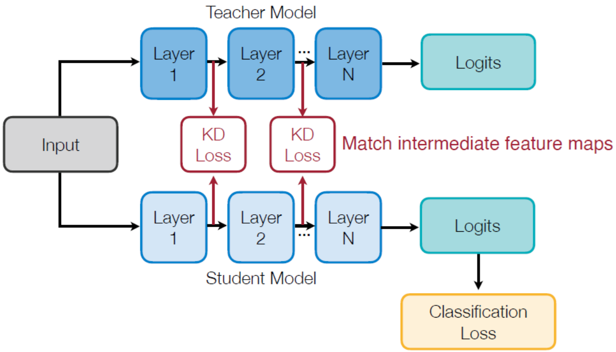
- **Distillation loss:** squared error/cross-entropy between teacher and student features



Matching Features

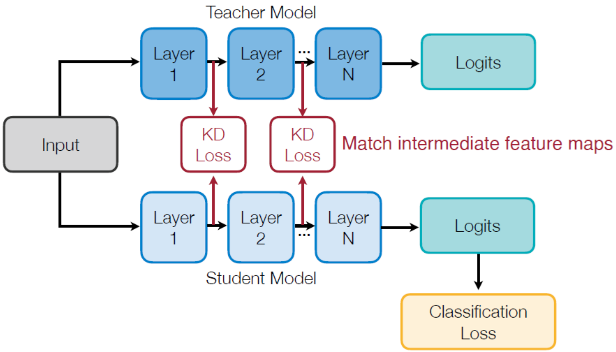
- **Distillation loss:** squared error/cross-entropy between teacher and student features

- **Problem:** feature sizes don't match!



Matching Features

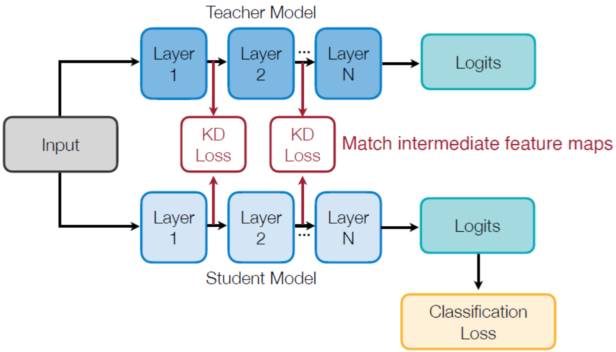
- **Distillation loss:** squared error/cross-entropy between teacher and student features
- **Problem:** feature sizes don't match!
- **Solution:** multiply with a learnable projection matrix to align sizes [1]



[1] A. Romero, N. Ballas, S. Ebrahimi Kahou, A. Chassang, C. Gatta, Y. Bengio, "FitNets: Hints for Thin Deep Nets," ICLR 2015

Matching Features

- **Distillation loss:** squared error/cross-entropy between teacher and student features
- **Problem:** feature sizes don't match!
- **Solution:** multiply with a learnable projection matrix to align sizes [1]



- It is also possible to match weights, gradients, sparsity patterns, and more [2]

[1] A. Romero, N. Ballas, S. Ebrahimi Kahou, A. Chassang, C. Gatta, Y. Bengio, "FitNets: Hints for Thin Deep Nets," ICLR 2015

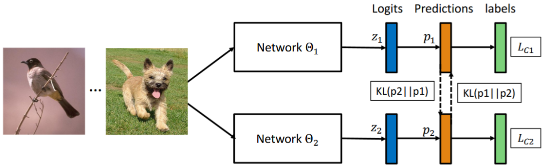
[2] J. Gou, B. Yu, S. J. Maybank, D. Tao, "Knowledge Distillation: A Survey," IJCV 2021

Online Distillation

- **Typical KD** needs pre-trained teacher:
  1. Large $\rightarrow$ expensive
  2. Fixed $\rightarrow$ inflexible

Online Distillation

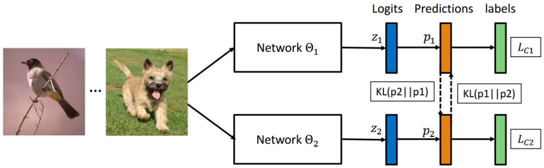
- **Typical KD** needs pre-trained teacher:
  1. Large $\rightarrow$ expensive
  2. Fixed $\rightarrow$ inflexible
- **Online distillation**:
  1. Students teach each other!



Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, "Deep Mutual Learning," CVPR 2018

Online Distillation

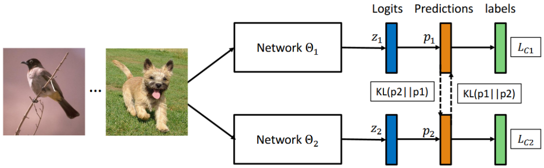
- **Typical KD** needs pre-trained teacher:
  1. Large $\rightarrow$ expensive
  2. Fixed $\rightarrow$ inflexible
- **Online distillation**:
  1. Students teach each other!
  2. Networks can be arbitrary
  3. Both have class. & distillation loss



Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, “Deep Mutual Learning,” CVPR 2018

Online Distillation

- **Typical KD** needs pre-trained teacher:
  1. Large $\rightarrow$ expensive
  2. Fixed $\rightarrow$ inflexible
- **Online distillation**:
  1. Students teach each other!
  2. Networks can be arbitrary
  3. Both have class. & distillation loss

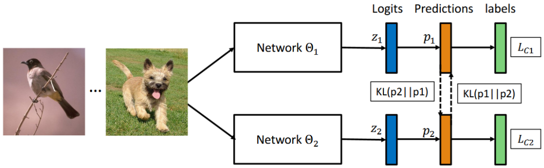


Dataset	Network Types		Independent	
	Net1	Net 2	Net 1	Net 2
CIFAR-100	WRN-28-10	ResNet-32	78.69	68.99
	MobilNet	ResNet-32	73.65	68.99

Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, “Deep Mutual Learning,” CVPR 2018

Online Distillation

- **Typical KD** needs pre-trained teacher:
  1. Large → expensive
  2. Fixed → inflexible
- **Online distillation**:
  1. Students teach each other!
  2. Networks can be arbitrary
  3. Both have class. & distillation loss

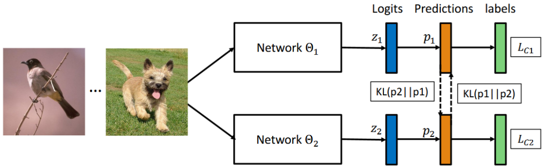


Dataset	Network Types Net1	Net 2	Independent Net 1	Net 2	1 distills 2 Net 2
CIFAR-100	WRN-28-10	ResNet-32	78.69	68.99	69.48
	MobilNet	ResNet-32	73.65	68.99	69.12

Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, "Deep Mutual Learning," CVPR 2018

Online Distillation

- **Typical KD** needs pre-trained teacher:
  1. Large → expensive
  2. Fixed → inflexible
- **Online distillation**:
  1. Students teach each other!
  2. Networks can be arbitrary
  3. Both have class. & distillation loss

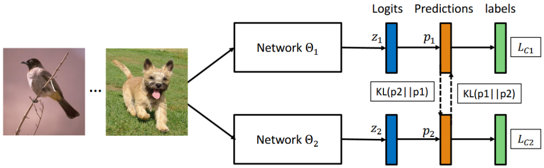


Dataset	Network Types		Independent		1 distills 2	DML	
	Net1	Net 2	Net 1	Net 2	Net 2	Net 1	Net 2
CIFAR-100	WRN-28-10	ResNet-32	78.69	68.99	69.48	78.96	70.73
	MobilNet	ResNet-32	73.65	68.99	69.12	76.13	71.10

Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, "Deep Mutual Learning," CVPR 2018

Online Distillation

- **Typical KD** needs pre-trained teacher:
  1. Large → expensive
  2. Fixed → inflexible
- **Online distillation**:
  1. Students teach each other!
  2. Networks can be arbitrary
  3. Both have class. & distillation loss



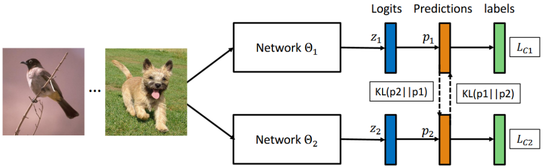
Dataset	Network Types		Independent		1 distills 2	DML	
	Net1	Net 2	Net 1	Net 2	Net 2	Net 1	Net 2
CIFAR-100	WRN-28-10	ResNet-32	78.69	68.99	69.48	78.96	70.73
	MobilNet	ResNet-32	73.65	68.99	69.12	76.13	71.10

- Online distillation leads to better performance **for both networks**!

Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, "Deep Mutual Learning," CVPR 2018

Online Distillation

- **Typical KD** needs pre-trained teacher:
  1. Large → expensive
  2. Fixed → inflexible
- **Online distillation**:
  1. Students teach each other!
  2. Networks can be arbitrary
  3. Both have class. & distillation loss



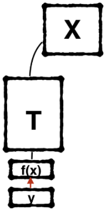
Dataset	Network Types		Independent		1 distills 2	DML	
	Net1	Net 2	Net 1	Net 2	Net 2	Net 1	Net 2
CIFAR-100	WRN-28-10	ResNet-32	78.69	68.99	69.48	78.96	70.73
	MobilNet	ResNet-32	73.65	68.99	69.12	76.13	71.10

- Online distillation leads to better performance **for both networks**!
- **Intuition:** Different prior knowledge (i.e., different initializations) → different soft labels

Y. Zhang, T. Xiang, T. M. Hospedales, H. Lu, "Deep Mutual Learning," CVPR 2018

Self-Distillation

- A single neural network [1]:
    1. **Step 0**: train only on dataset

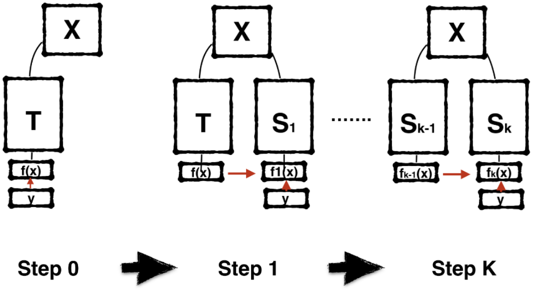


Step 0

[1] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, A. Anandkumar, "Born Again Neural Networks," ICML 2018

Self-Distillation

- A single neural network [1]:
  1. **Step 0**: train only on dataset
  2. **Step k**: train on dataset and soft labels from network at step $k-1$

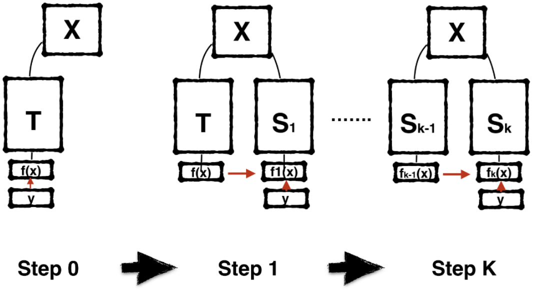


[1] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, A. Anandkumar, “Born Again Neural Networks,” ICML 2018

Self-Distillation

- A single neural network [1]:
  1. **Step 0**: train only on dataset
  2. **Step k**: train on dataset and soft labels from network at step $k-1$

- A form of **self-regularization**!

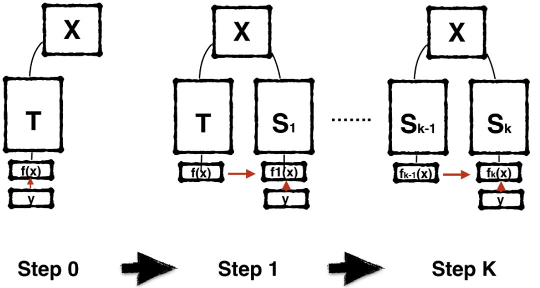


[1] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, A. Anandkumar, "Born Again Neural Networks," ICML 2018

TU/e

Self-Distillation

- A single neural network [1]:
    1. **Step 0**: train only on dataset
    2. **Step k**: train on dataset and soft labels from network at step $k-1$



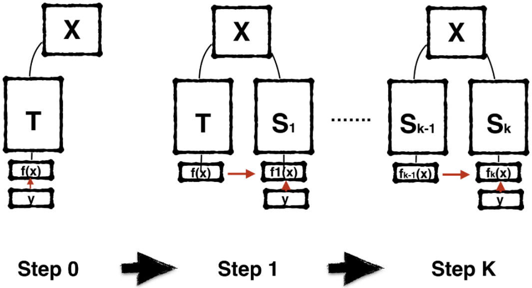
- A form of **self-regularization**!

- **Alternative:** Split network into parts, train early parts with output of later parts with KD

[1] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, A. Anandkumar, "Born Again Neural Networks," ICML 2018

Self-Distillation

- A single neural network [1]:
  1. **Step 0**: train only on dataset
  2. **Step k**: train on dataset and soft labels from network at step $k-1$



- A form of **self-regularization**!
- **Alternative:** Split network into parts, train early parts with output of later parts with KD
- **Advantage:** Can use only subset of network for performance-complexity trade-offs [2]

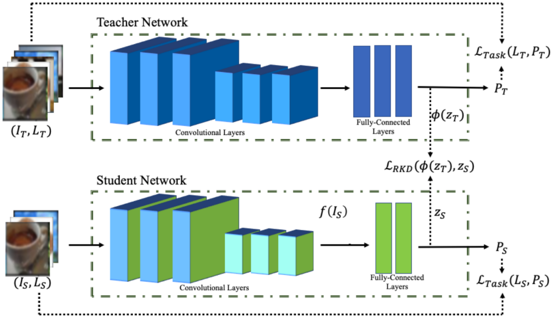
[1] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, A. Anandkumar, "Born Again Neural Networks," ICML 2018
[2] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, K. Ma, "Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation," ICCV 2019

Distillation for Task Specialization

- So far, teacher and student were trained for the same task.

Distillation for Task Specialization

- So far, teacher and student were trained for the same task.
- But **student specialization** is also possible.

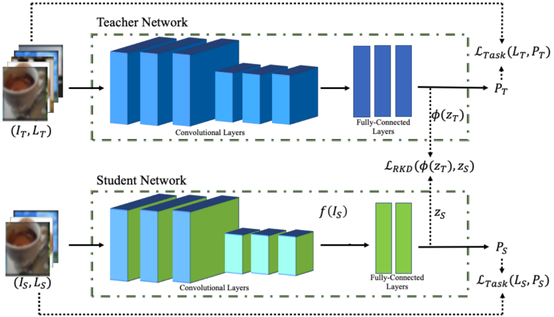


[1] L.-Y. Wang, A. Rhodes, W.-C. Feng, “Class Specialized Knowledge Distillation,” ACCV 2022

Distillation for Task Specialization

- So far, teacher and student were trained for the same task.
- But **student specialization** is also possible.

- **Problem:** different number of classes for teacher & student!

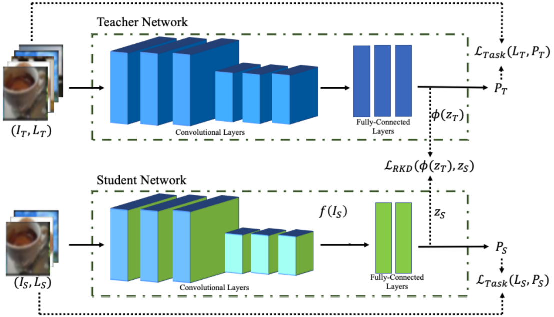


[1] L.-Y. Wang, A. Rhodes, W.-C. Feng, "Class Specialized Knowledge Distillation," ACCV 2022

TU/e

Distillation for Task Specialization

- So far, teacher and student were trained for the same task.
- But **student specialization** is also possible.



- **Problem:** different number of classes for teacher & student!

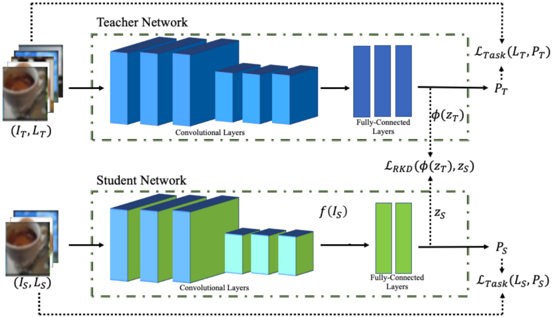
- Train with subset of dataset and all outputs?

[1] L.-Y. Wang, A. Rhodes, W.-C. Feng, "Class Specialized Knowledge Distillation," ACCV 2022

TU/e

Distillation for Task Specialization

- So far, teacher and student were trained for the same task.
- But **student specialization** is also possible.



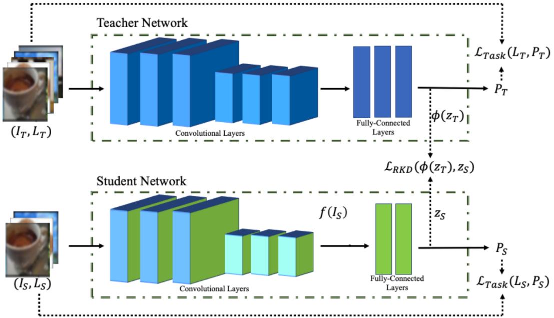
- **Problem:** different number of classes for teacher & student!

- Train with subset of dataset and all outputs? **Dark knowledge** uses student capacity!

[1] L.-Y. Wang, A. Rhodes, W.-C. Feng, "Class Specialized Knowledge Distillation," ACCV 2022

Distillation for Task Specialization

- So far, teacher and student were trained for the same task.
- But **student specialization** is also possible.
- **Problem:** different number of classes for teacher & student!



- Train with subset of dataset and all outputs? **Dark knowledge** uses student capacity!
- **Better solution:**
  1. Take a subset of teacher outputs z_T (denoted $\phi(z_T)$ in [1]) to align with z_S
  2. Renormalize $\phi(z_T)$ with temperature-based softmax and apply distillation loss

[1] L.-Y. Wang, A. Rhodes, W.-C. Feng, “Class Specialized Knowledge Distillation,” ACCV 2022

Conclusion

Summary:

- **Knowledge distillation** is transfer of knowledge from one network to another.
- Can be from teacher to student, or from student to itself.
- Also useful for task specialization.

Conclusion

Summary:

- **Knowledge distillation** is transfer of knowledge from one network to another.
- Can be from teacher to student, or from student to itself.
- Also useful for task specialization.

Next Time (After Carnival):

- NXP guest lecture (dr Willem Sandberg): Neural architecture search

TU/e