Knowledge Distillation : transfert de connaissances entre deux réseaux à l'entraînement

Course organisation

Sessions

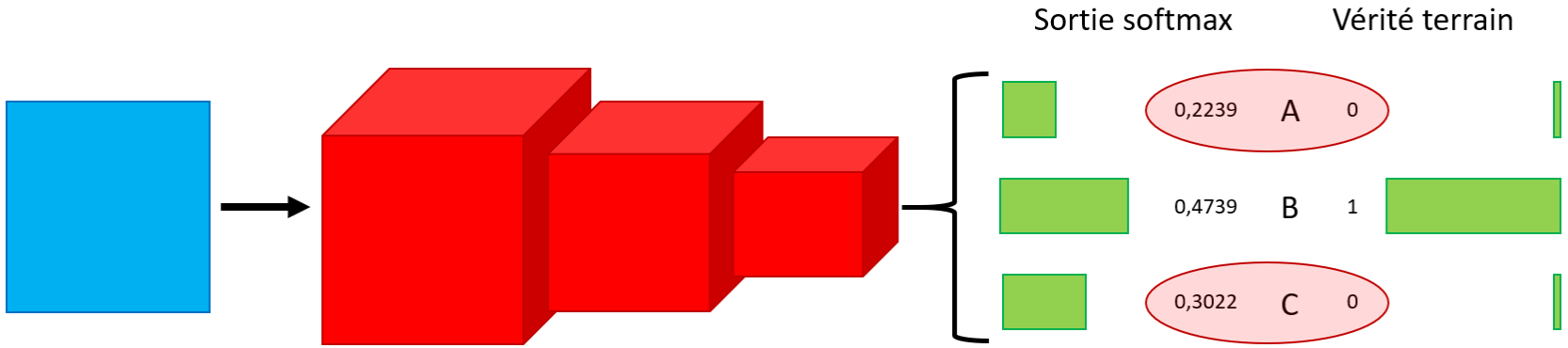
1. Deep Learning and Transfer Learning,
2. Quantification,
3. Pruning,
4. Factorization,
5. Fact. pt.2 : Operators and Architectures,
6. Distillation,
7. Embedded Software and Hardware for DL.
8. Presentations for challenge.

Course organisation

Sessions

1. Deep Learning and Transfer Learning,
2. Quantification,
3. Pruning,
4. Factorization,
5. Fact. pt.2 : Operators and Architectures,
6. Distillation,
7. Embedded Software and Hardware for DL.
8. Presentations for challenge.

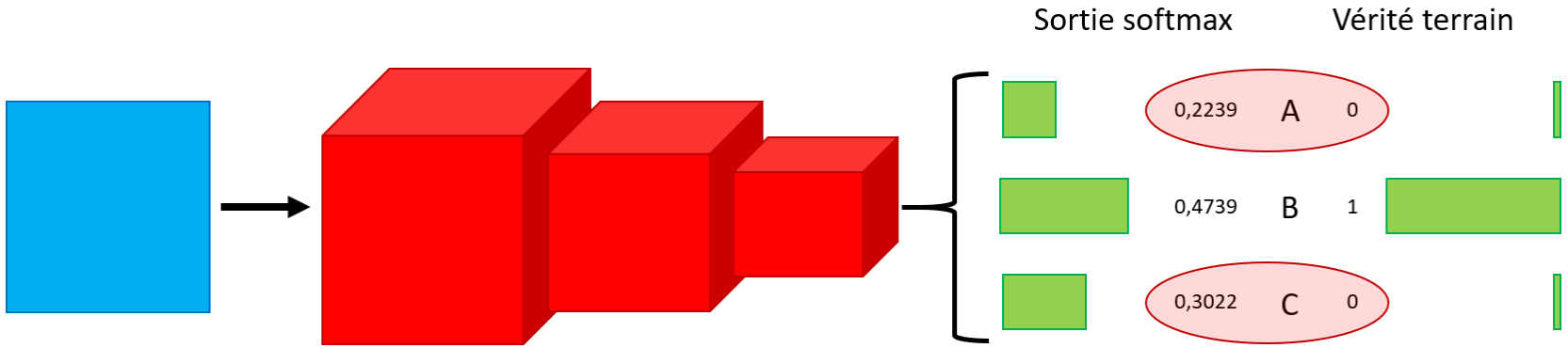
Observation originale



Distilling the Knowledge in a Neural Network, Hinton & al. 2015

- Les valeurs des classes fausses renseignent sur la généralisation du réseau
- Ces labels doux peuvent servir de labels plus pertinents pour entraîner un autre réseau

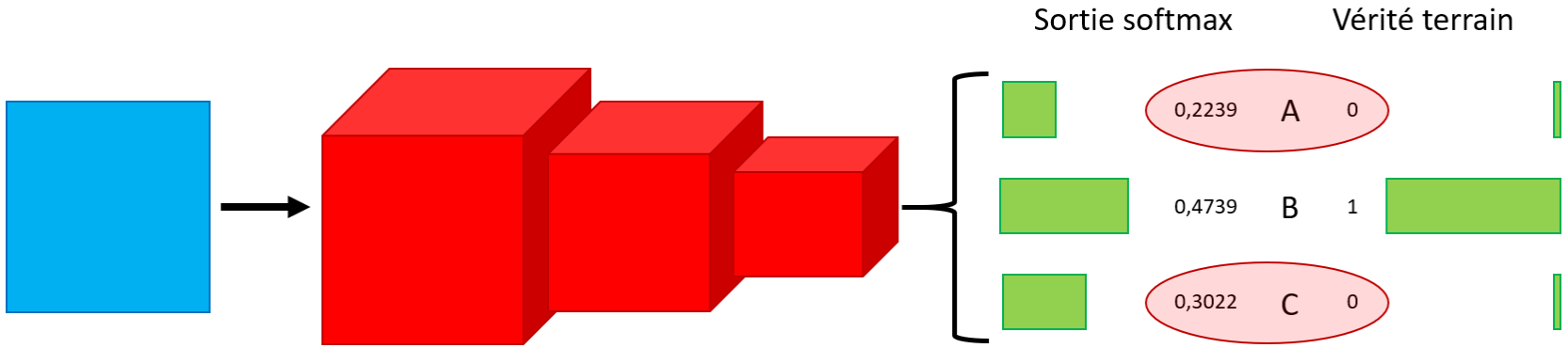
Observation originale



Distilling the Knowledge in a Neural Network, Hinton & al. 2015

- Les valeurs des classes fausses renseignent sur la généralisation du réseau
- Ces labels doux peuvent servir de labels plus pertinents pour entraîner un autre réseau

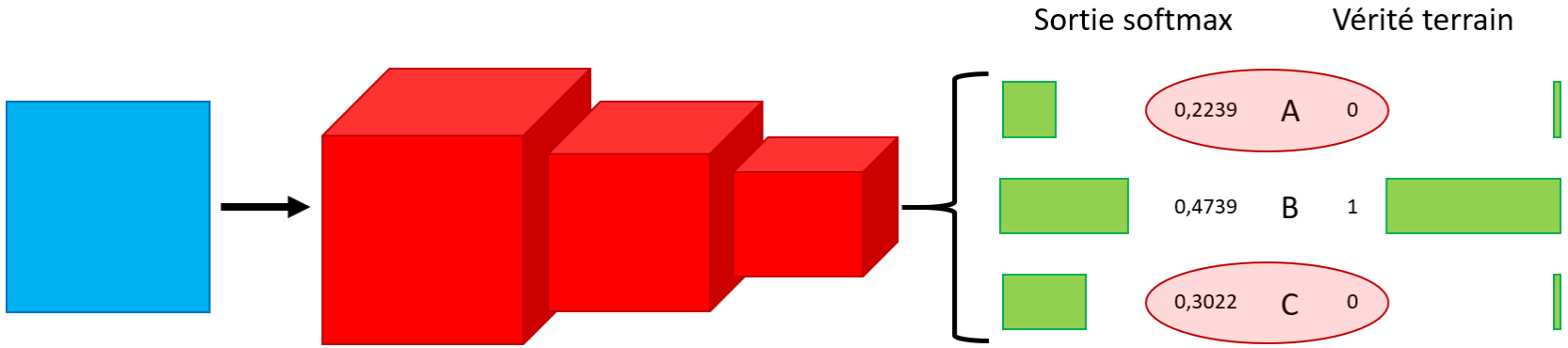
Observation originale



Distilling the Knowledge in a Neural Network, Hinton & al. 2015

- Les valeurs des classes fausses renseignent sur la généralisation du réseau
- Ces labels doux peuvent servir de labels plus pertinents pour entraîner un autre réseau

Observation originale



Distilling the Knowledge in a Neural Network, Hinton & al. 2015

- Les valeurs des classes fausses renseignent sur la généralisation du réseau
- Ces labels doux peuvent servir de labels plus pertinents pour entraîner un autre réseau

Principe de base

Distillation de Hinton

$$\mathcal{L}_{KD} = \underbrace{H(y_{true}, P_s)}_{\text{terme supervisé}} + \underbrace{\lambda D_{KL}(P_T, P_S)}_{\text{terme de distillation}}$$

avec P_S la sortie de l'élève, P_T la sortie du professeur, H l'entropie croisée et D_{KL} la divergence de Kullback-Leibler (ou entropie relative).

Considérons les sorties *softmax* : $q_i = \frac{e^{z_i/T}}{\sum_i e^{z_j/T}}$

- T est choisi égal à 1 pour l'inférence mais supérieur à 1 pour le terme de distillation (les sorties sont alors plus douces)
- Comme les sorties sont alors d'amplitude $1/T^2$, il faut multiplier le résultat par T^2

Principe de base

Distillation de Hinton

$$\mathcal{L}_{KD} = \underbrace{H(y_{true}, P_s)}_{\text{terme supervisé}} + \underbrace{\lambda D_{KL}(P_T, P_S)}_{\text{terme de distillation}}$$

avec P_S la sortie de l'élève, P_T la sortie du professeur, H l'entropie croisée et D_{KL} la divergence de Kullback-Leibler (ou entropie relative).

Considérons les sorties *softmax* : $q_i = \frac{e^{z_i/T}}{\sum_i e^{z_j/T}}$

- T est choisi égal à 1 pour l'inférence mais supérieur à 1 pour le terme de distillation (les sorties sont alors plus douces)
- Comme les sorties sont alors d'amplitude $1/T^2$, il faut multiplier le résultat par T^2

Principe de base

Distillation de Hinton

$$\mathcal{L}_{KD} = \underbrace{H(y_{true}, P_s)}_{\text{terme supervisé}} + \underbrace{\lambda D_{KL}(P_T, P_S)}_{\text{terme de distillation}}$$

avec P_S la sortie de l'élève, P_T la sortie du professeur, H l'entropie croisée et D_{KL} la divergence de Kullback-Leibler (ou entropie relative).

Considérons les sorties *softmax* : $q_i = \frac{e^{z_i/T}}{\sum_i e^{z_j/T}}$

- T est choisi égal à 1 pour l'inférence mais supérieur à 1 pour le terme de distillation (les sorties sont alors plus douces)
- Comme les sorties sont alors d'amplitude $1/T^2$, il faut multiplier le résultat par T^2

Principe de base

Distillation de Hinton

$$\mathcal{L}_{KD} = \underbrace{H(y_{true}, P_s)}_{\text{terme supervisé}} + \underbrace{\lambda D_{KL}(P_T, P_S)}_{\text{terme de distillation}}$$

avec P_S la sortie de l'élève, P_T la sortie du professeur, H l'entropie croisée et D_{KL} la divergence de Kullback-Leibler (ou entropie relative).

Considérons les sorties *softmax* : $q_i = \frac{e^{z_i/T}}{\sum_i e^{z_j/T}}$

- T est choisi égal à 1 pour l'inférence mais supérieur à 1 pour le terme de distillation (les sorties sont alors plus douces)
- Comme les sorties sont alors d'amplitude $1/T^2$, il faut multiplier le résultat par T^2

Utilisations

- Compression de réseaux de neurones (*LIT: Block-wise intermediate representation training for model compression*, Koratana & al., 2018)
- Self-distillation (*Born-Again Neural Networks*, Furlanello & al., 2018)
- Transfert de connaissances sur des données absentes (*Distilling the Knowledge in a Neural Network*, Hinton & al. 2015)

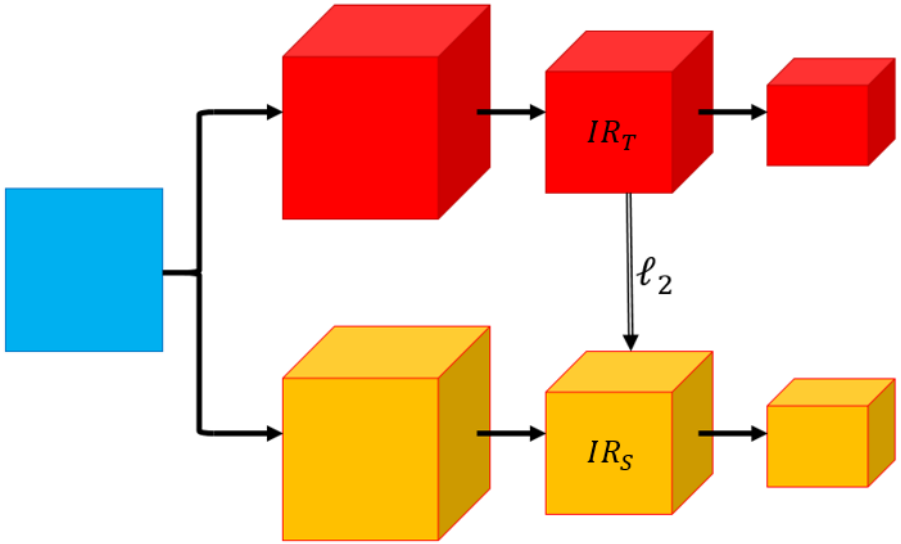
Utilisations

- Compression de réseaux de neurones (*LIT: Block-wise intermediate representation training for model compression*, Koratana & al., 2018)
- Self-distillation (*Born-Again Neural Networks*, Furlanello & al., 2018)
- Transfert de connaissances sur des données absentes (*Distilling the Knowledge in a Neural Network*, Hinton & al. 2015)

Utilisations

- Compression de réseaux de neurones (*LIT: Block-wise intermediate representation training for model compression*, Koratana & al., 2018)
- Self-distillation (*Born-Again Neural Networks*, Furlanello & al., 2018)
- Transfert de connaissances sur des données absentes (*Distilling the Knowledge in a Neural Network*, Hinton & al. 2015)

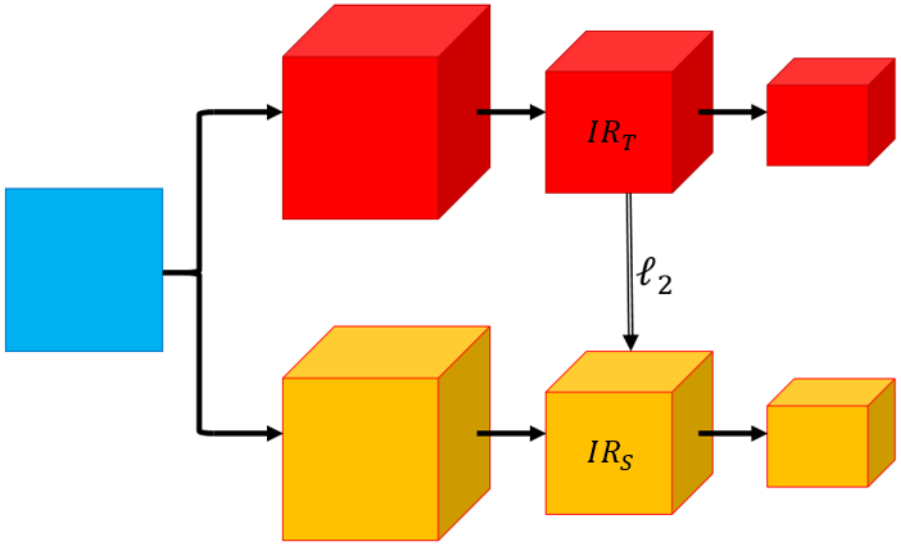
L'étape d'après : la *features distillation* 1/2



Distillation sur les représentations intermédiaires (*FitNets : hints for thin deep nets*, Romero & al., 2014)

- Une ou plusieurs représentations intermédiaires de l'élève doivent imiter celles du professeur
- La distance entre ces deux représentations intermédiaires est, le plus souvent, la norme ℓ_2
- Ce nouveau terme, $\mathcal{L}_{IR} = \|IR_T - IR_S\|_2$, s'ajoute à $\mathcal{L}_{KD}$

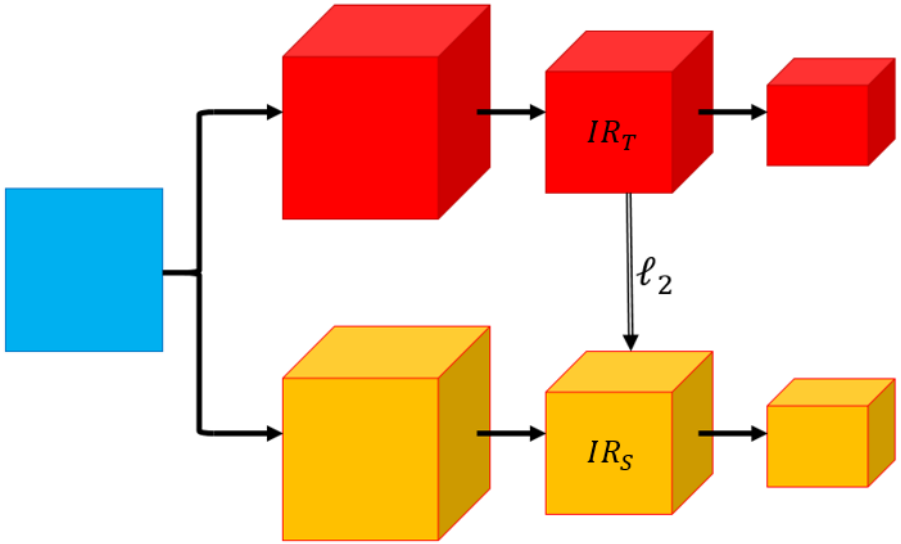
L'étape d'après : la *features distillation* 1/2



Distillation sur les représentations intermédiaires (*FitNets : hints for thin deep nets*, Romero & al., 2014)

- Une ou plusieurs représentations intermédiaires de l'élève doivent imiter celles du professeur
- La distance entre ces deux représentations intermédiaires est, le plus souvent, la norme ℓ_2
- Ce nouveau terme, $\mathcal{L}_{IR} = \|IR_T - IR_S\|_2$, s'ajoute à $\mathcal{L}_{KD}$

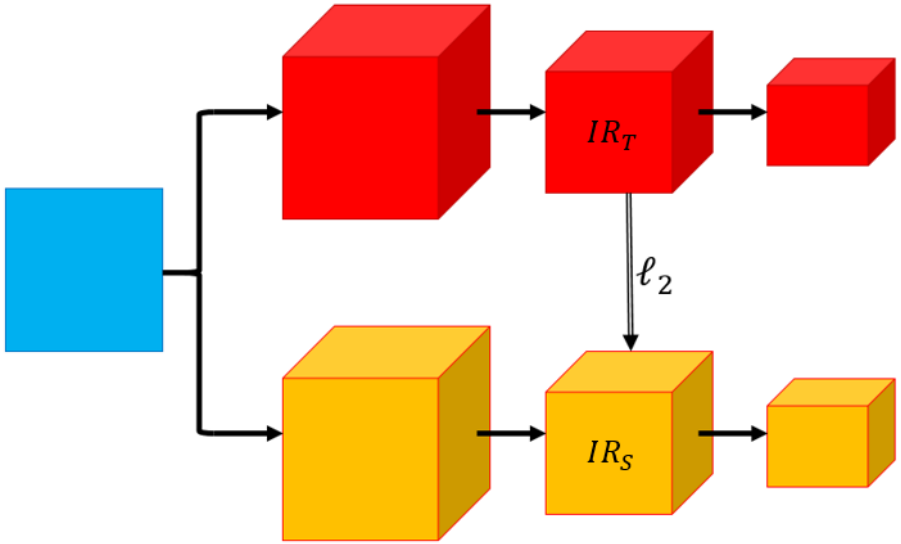
L'étape d'après : la *features distillation* 1/2



Distillation sur les représentations intermédiaires (*FitNets : hints for thin deep nets*, Romero & al., 2014)

- Une ou plusieurs représentations intermédiaires de l'élève doivent imiter celles du professeur
- La distance entre ces deux représentations intermédiaires est, le plus souvent, la norme ℓ_2
- Ce nouveau terme, $\mathcal{L}_{IR} = \|IR_T - IR_S\|_2$, s'ajoute à $\mathcal{L}_{KD}$

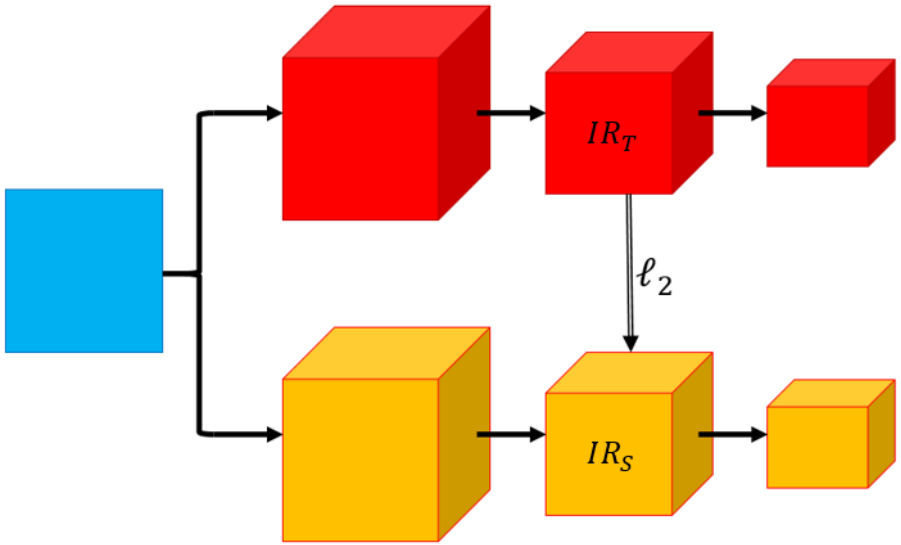
L'étape d'après : la *features distillation* 1/2



Distillation sur les représentations intermédiaires (*FitNets : hints for thin deep nets*, Romero & al., 2014)

- Une ou plusieurs représentations intermédiaires de l'élève doivent imiter celles du professeur
- La distance entre ces deux représentations intermédiaires est, le plus souvent, la norme ℓ_2
- Ce nouveau terme, $\mathcal{L}_{IR} = \|IR_T - IR_S\|_2$, s'ajoute à $\mathcal{L}_{KD}$

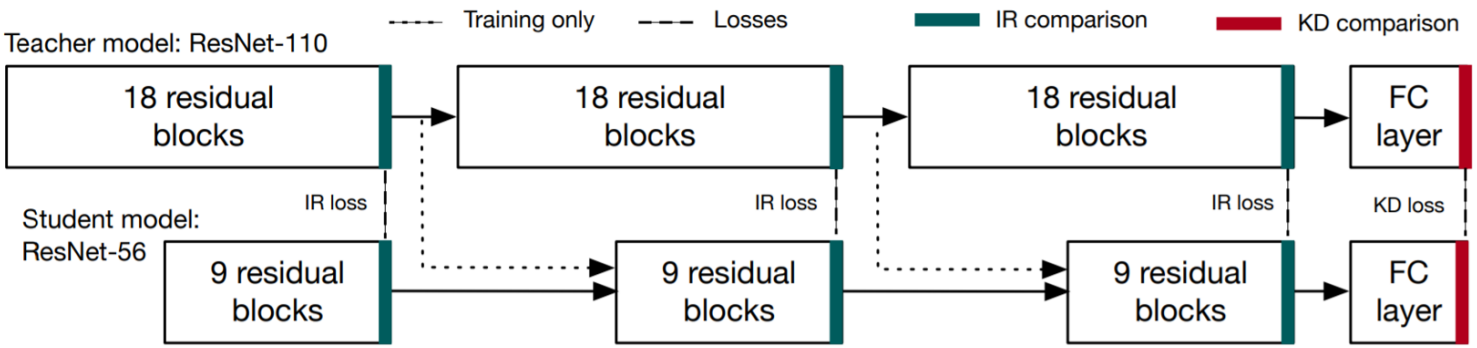
L'étape d'après : la *features distillation* 1/2



Distillation sur les représentations intermédiaires (*FitNets : hints for thin deep nets*, Romero & al., 2014)

- Une ou plusieurs représentations intermédiaires de l'élève doivent imiter celles du professeur
- La distance entre ces deux représentations intermédiaires est, le plus souvent, la norme ℓ_2
- Ce nouveau terme, $\mathcal{L}_{IR} = \| IR_T - IR_S \|_2$, s'ajoute à $\mathcal{L}_{KD}$

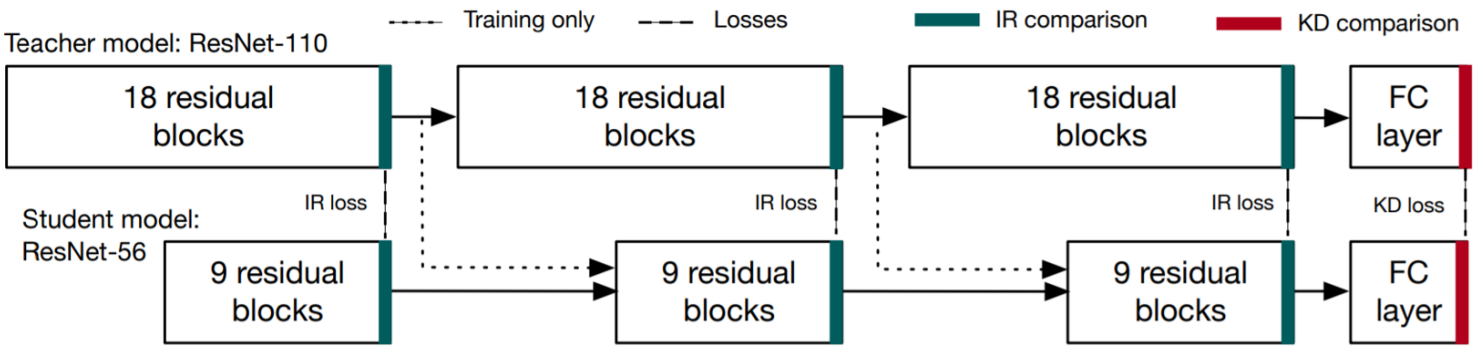
L'étape d'après : la *features distillation* 2/2



LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

- Représentations intermédiaires en sortie de chaque bloc
- Entraînement avec en entrée les sorties intermédiaires du professeur (spécificité de LIT)
- Les représentations intermédiaires comparées doivent être de même dimension
- Un réseau régresseur (une couche dense ou une convolution 1×1) est inséré pour faire correspondre les dimensions

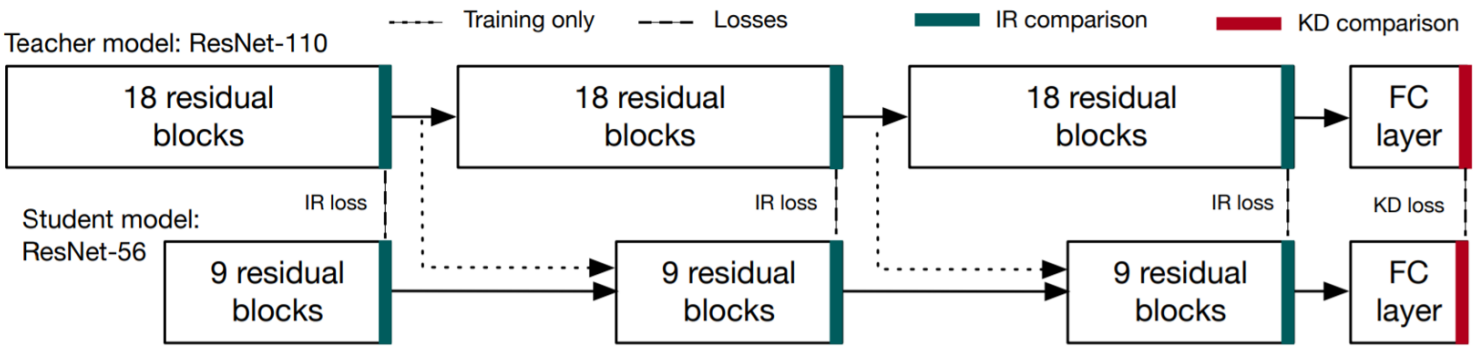
L'étape d'après : la *features distillation* 2/2



LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

- Représentations intermédiaires en sortie de chaque bloc
- Entraînement avec en entrée les sorties intermédiaires du professeur (spécificité de LIT)
- Les représentations intermédiaires comparées doivent être de même dimension
- Un réseau régresseur (une couche dense ou une convolution 1×1) est inséré pour faire correspondre les dimensions

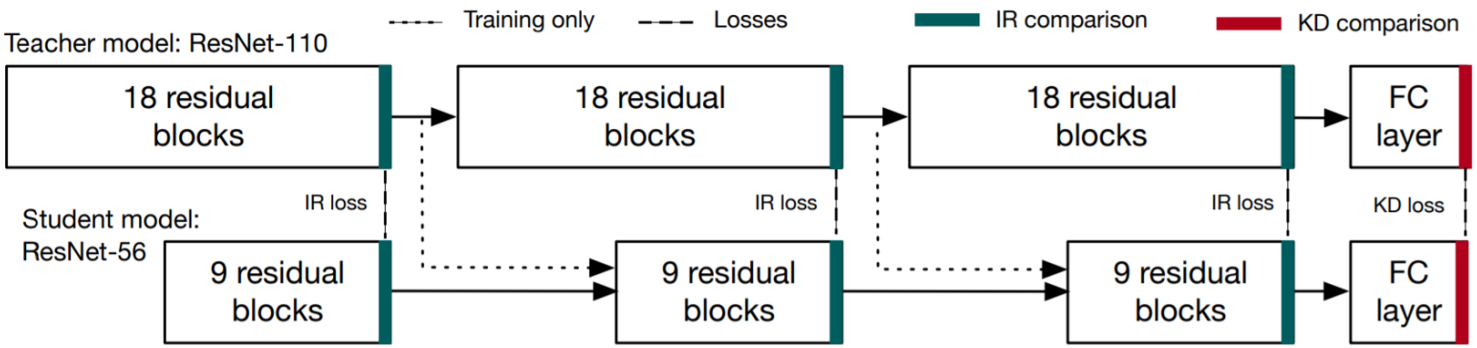
L'étape d'après : la *features distillation* 2/2



LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

- Représentations intermédiaires en sortie de chaque bloc
- Entraînement avec en entrée les sorties intermédiaires du professeur (spécificité de LIT)
- Les représentations intermédiaires comparées doivent être de même dimension
- Un réseau régresseur (une couche dense ou une convolution 1×1) est inséré pour faire correspondre les dimensions

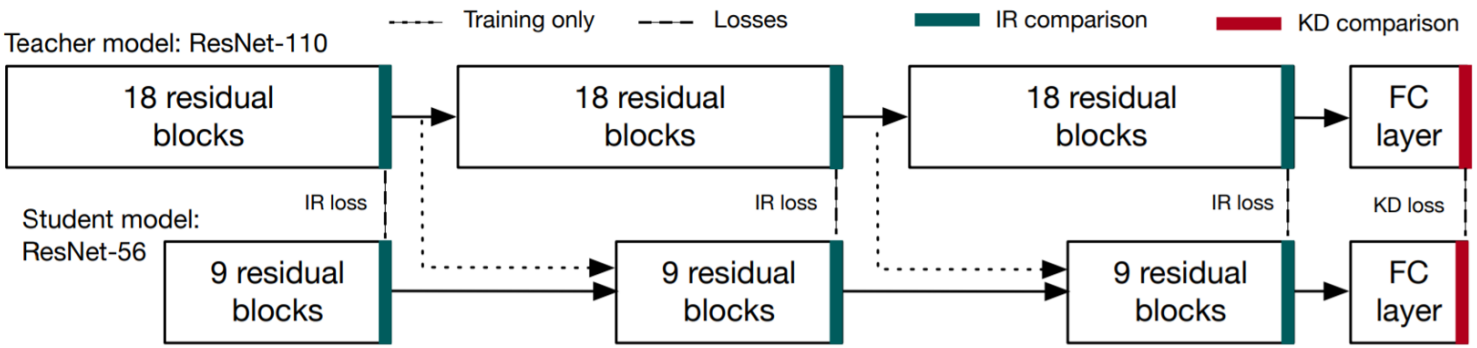
L'étape d'après : la *features distillation* 2/2



LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

- Représentations intermédiaires en sortie de chaque bloc
- Entraînement avec en entrée les sorties intermédiaires du professeur (spécificité de LIT)
- Les représentations intermédiaires comparées doivent être de même dimension
- Un réseau régresseur (une couche dense ou une convolution 1×1) est inséré pour faire correspondre les dimensions

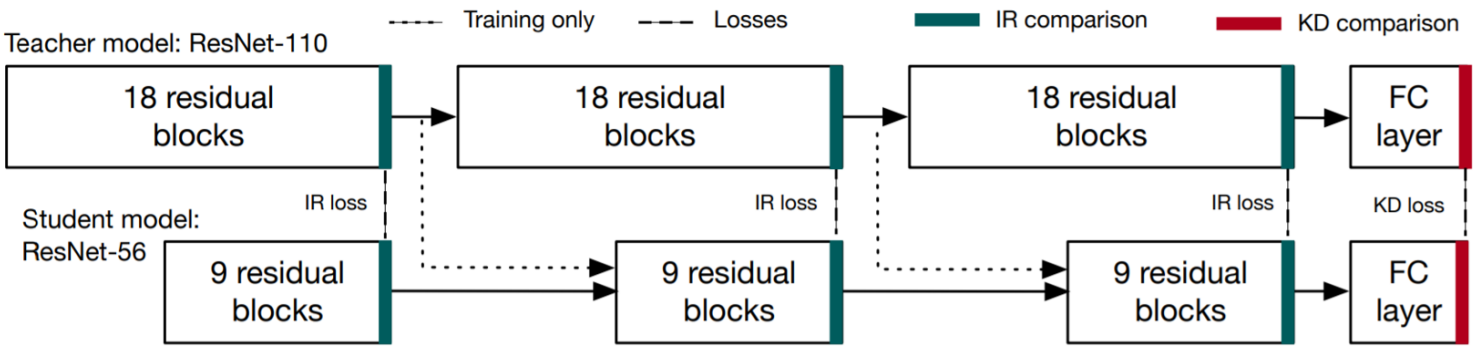
L'étape d'après : la *features distillation* 2/2



LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

- Représentations intermédiaires en sortie de chaque bloc
- Entraînement avec en entrée les sorties intermédiaires du professeur (spécificité de LIT)
- Les représentations intermédiaires comparées doivent être de même dimension
- Un réseau régresseur (une couche dense ou une convolution 1×1) est inséré pour faire correspondre les dimensions

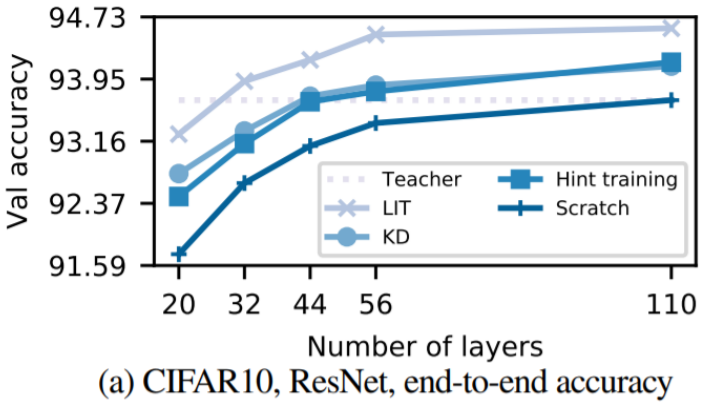
L'étape d'après : la *features distillation* 2/2



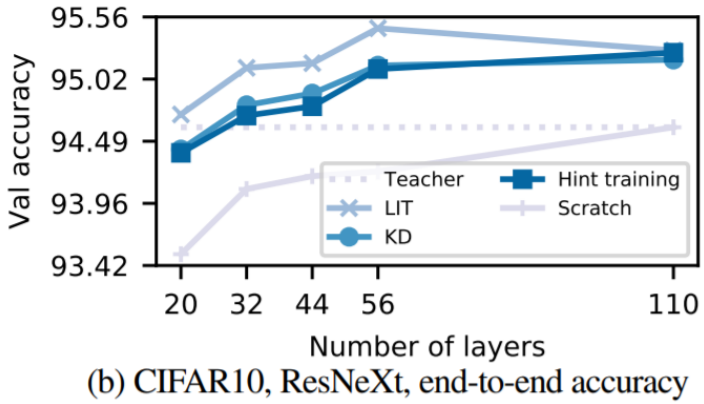
LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

- Représentations intermédiaires en sortie de chaque bloc
- Entraînement avec en entrée les sorties intermédiaires du professeur (spécificité de LIT)
- Les représentations intermédiaires comparées doivent être de même dimension
- Un réseau régresseur (une couche dense ou une convolution 1×1) est inséré pour faire correspondre les dimensions

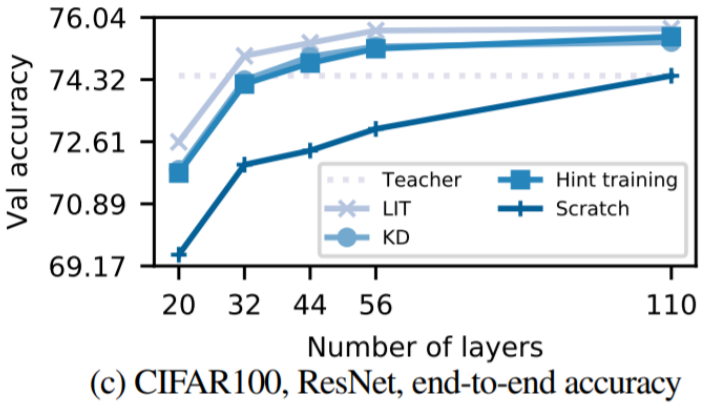
Quelques résultats



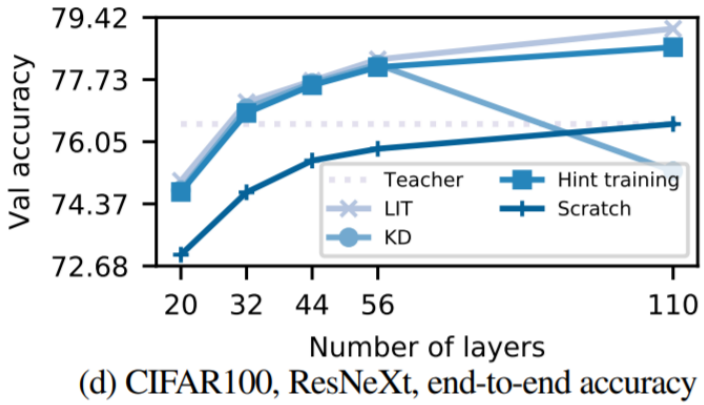
(a) CIFAR10, ResNet, end-to-end accuracy



(b) CIFAR10, ResNeXt, end-to-end accuracy



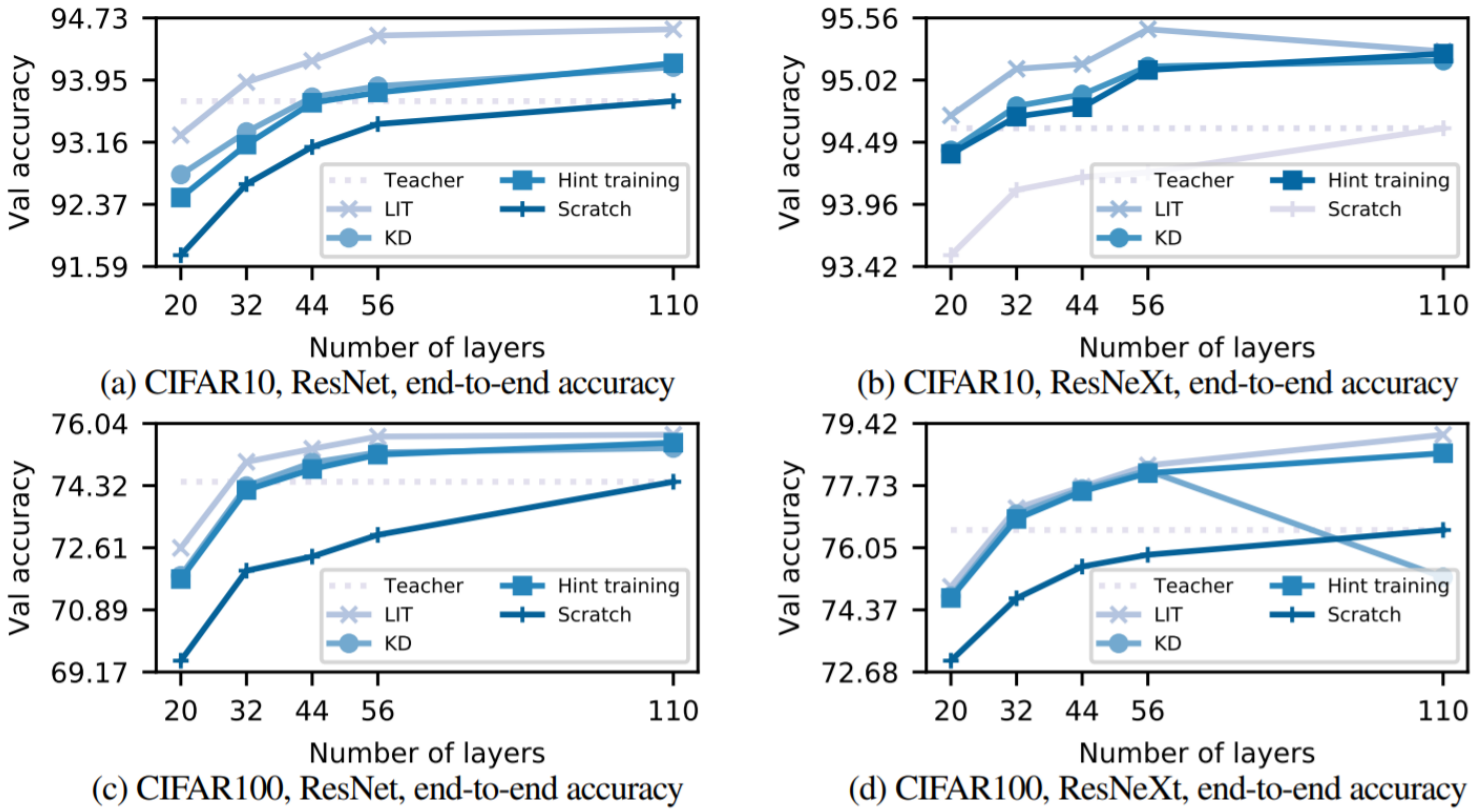
(c) CIFAR100, ResNet, end-to-end accuracy



(d) CIFAR100, ResNeXt, end-to-end accuracy

LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

Quelques résultats



(a) CIFAR10, ResNet, end-to-end accuracy

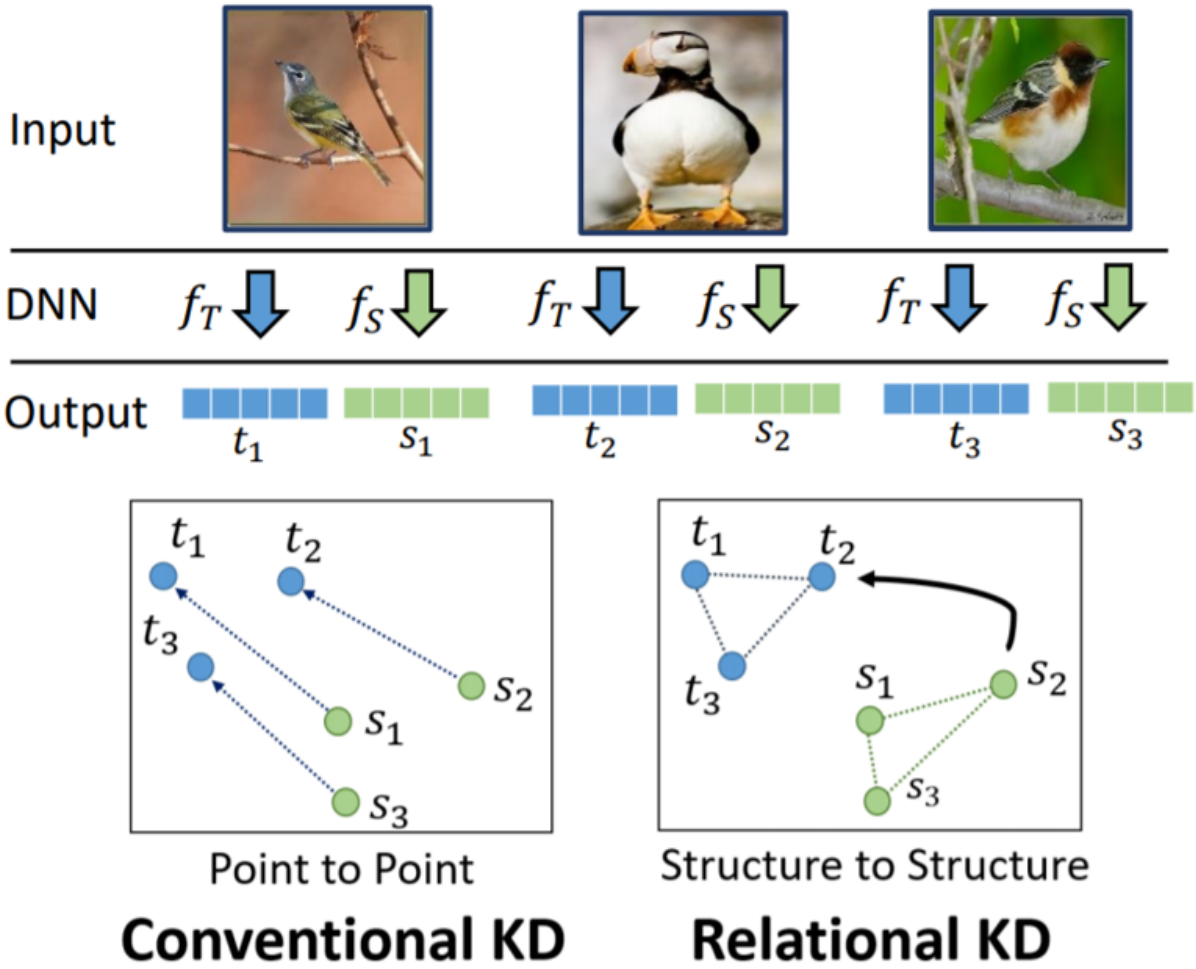
(b) CIFAR10, ResNeXt, end-to-end accuracy

(c) CIFAR100, ResNet, end-to-end accuracy

(d) CIFAR100, ResNeXt, end-to-end accuracy

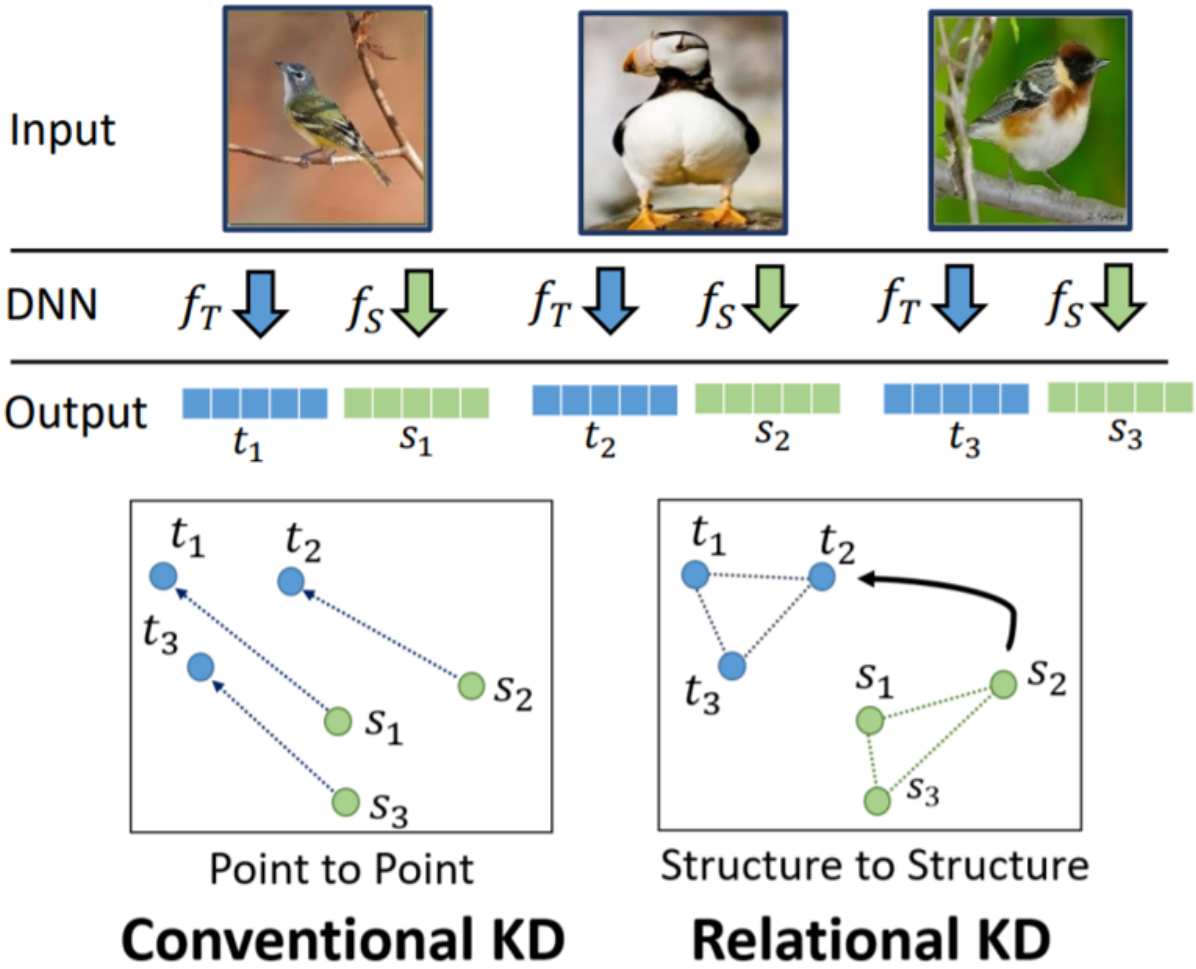
LIT: Block-wise intermediate representation training for model compression, Koratana & al., 2018

RKD : apprendre comment discriminer les données 1/2



Relational Knowledge Distillation, Park & al., 2019

RKD : apprendre comment discriminer les données 1/2



Relational Knowledge Distillation, Park & al., 2019

RKD : apprendre comment discriminer les données 2/2

Relational Knowledge Distillation

$\mathcal{L}_{RKD} = \sum_{i,j \in \mathcal{X}^N} \ell(\phi(t_i, t_j), \phi(s_i, s_j))$ avec $\phi(t_i, t_j) = \frac{1}{\mu} \| t_i - t_j \|_2$,
ℓ la norme de Huber, alias "norme ℓ_1 douce", μ un terme de normalisation et $\mathcal{X}^N$ le batch d'entraînement

On s'est totalement abstrait de la dimension des représentations intermédiaires.

- Pour chaque batch, on calcule la norme ℓ_2 entre paires de représentations intermédiaires chez le professeur et chez l'élève séparément
- On compare ces distances chez l'élève et chez le professeur
- On ajoute $\mathcal{L}_{RKD}$ à la loss

RKD : apprendre comment discriminer les données 2/2

Relational Knowledge Distillation

$\mathcal{L}_{RKD} = \sum_{i,j \in \mathcal{X}^N} \ell(\phi(t_i, t_j), \phi(s_i, s_j))$ avec $\phi(t_i, t_j) = \frac{1}{\mu} \| t_i - t_j \|_2$,
ℓ la norme de Huber, alias "norme ℓ_1 douce", μ un terme de normalisation et $\mathcal{X}^N$ le batch d'entraînement

On s'est totalement abstrait de la dimension des représentations intermédiaires.

- Pour chaque batch, on calcule la norme ℓ_2 entre paires de représentations intermédiaires chez le professeur et chez l'élève séparément
- On compare ces distances chez l'élève et chez le professeur
- On ajoute $\mathcal{L}_{RKD}$ à la loss

RKD : apprendre comment discriminer les données 2/2

Relational Knowledge Distillation

$\mathcal{L}_{RKD} = \sum_{i,j \in \mathcal{X}^N} \ell(\phi(t_i, t_j), \phi(s_i, s_j))$ avec $\phi(t_i, t_j) = \frac{1}{\mu} \| t_i - t_j \|_2$,
ℓ la norme de Huber, alias "norme ℓ_1 douce", μ un terme de normalisation et $\mathcal{X}^N$ le batch d'entraînement

On s'est totalement abstrait de la dimension des représentations intermédiaires.

- Pour chaque batch, on calcule la norme ℓ_2 entre paires de représentations intermédiaires chez le professeur et chez l'élève séparément
- On compare ces distances chez l'élève et chez le professeur
- On ajoute $\mathcal{L}_{RKD}$ à la loss

RKD : apprendre comment discriminer les données 2/2

Relational Knowledge Distillation

$\mathcal{L}_{RKD} = \sum_{i,j \in \mathcal{X}^N} \ell(\phi(t_i, t_j), \phi(s_i, s_j))$ avec $\phi(t_i, t_j) = \frac{1}{\mu} \| t_i - t_j \|_2$,
ℓ la norme de Huber, alias "norme ℓ_1 douce", μ un terme de normalisation et $\mathcal{X}^N$ le batch d'entraînement

On s'est totalement abstrait de la dimension des représentations intermédiaires.

- Pour chaque batch, on calcule la norme ℓ_2 entre paires de représentations intermédiaires chez le professeur et chez l'élève séparément
- On compare ces distances chez l'élève et chez le professeur
- On ajoute $\mathcal{L}_{RKD}$ à la loss

RKD : apprendre comment discriminer les données 2/2

Relational Knowledge Distillation

$\mathcal{L}_{RKD} = \sum_{i,j \in \mathcal{X}^N} \ell(\phi(t_i, t_j), \phi(s_i, s_j))$ avec $\phi(t_i, t_j) = \frac{1}{\mu} \| t_i - t_j \|_2$,
ℓ la norme de Huber, alias "norme ℓ_1 douce", μ un terme de normalisation et $\mathcal{X}^N$ le batch d'entraînement

On s'est totalement abstrait de la dimension des représentations intermédiaires.

- Pour chaque batch, on calcule la norme ℓ_2 entre paires de représentations intermédiaires chez le professeur et chez l'élève séparément
- On compare ces distances chez l'élève et chez le professeur
- On ajoute $\mathcal{L}_{RKD}$ à la loss