Knowledge Distillation: A Survey

Jianping Gou[1] · **Baosheng Yu** [1] · **Stephen J. Maybank** [2] · **Dacheng Tao**[1]

Received: date / Accepted: date

Abstract In recent years, deep neural networks have been successful in both industry and academia, especially for computer vision tasks. The great success of deep learning is mainly due to its scalability to encode large-scale data and to maneuver billions of model parameters. However, it is [illegible] these cumbersome deep models

Keywords Deep neural networks · Model compression · Knowledge distillation · Knowledge transfer · Teacher-student architecture.

Knowledge Distillation: A Survey

Jianping Gou[1] · **Baosheng Yu** [1] · **Stephen J. Maybank** [2] · **Dacheng Tao**[1]

Intuition & various methods

Received: date / Accepted: date

Abstract In recent years, deep neural networks have been successful in both industry and academia, especially for computer vision tasks. The great success of deep learning is mainly due to its scalability to encode large-scale data and to maneuver billions of model parameters. However, it is a challenge to deploy these cumbersome deep models on devices with limited resources, *e.g.*, mobile phones

Keywords Deep neural networks · Model compression · Knowledge distillation · Knowledge transfer · Teacher-student architecture.

1 Introduction

Distilling the Knowledge in a Neural Network

Geoffrey Hinton[*†]
Google Inc.
Mountain View
geoffhinton@google.com

Oriol Vinyals[†]
Google Inc.
Mountain View
vinyals@google.com

Jeff Dean
Google Inc.
Mountain View
jeff@google.com

Abstract

A very simple way to improve the performance of almost any machine learning algorithm is to train many different models on the same data and then to average their predictions [3]. Unfortunately, making predictions using a whole ensemble of models is cumbersome and may be too computationally expensive to allow deployment to a large number of users, especially if the individual models are large neural nets. Caruana and his collaborators [1] have shown that it is possible to compress the knowledge in an ensemble into a single model which is much easier to dep compression

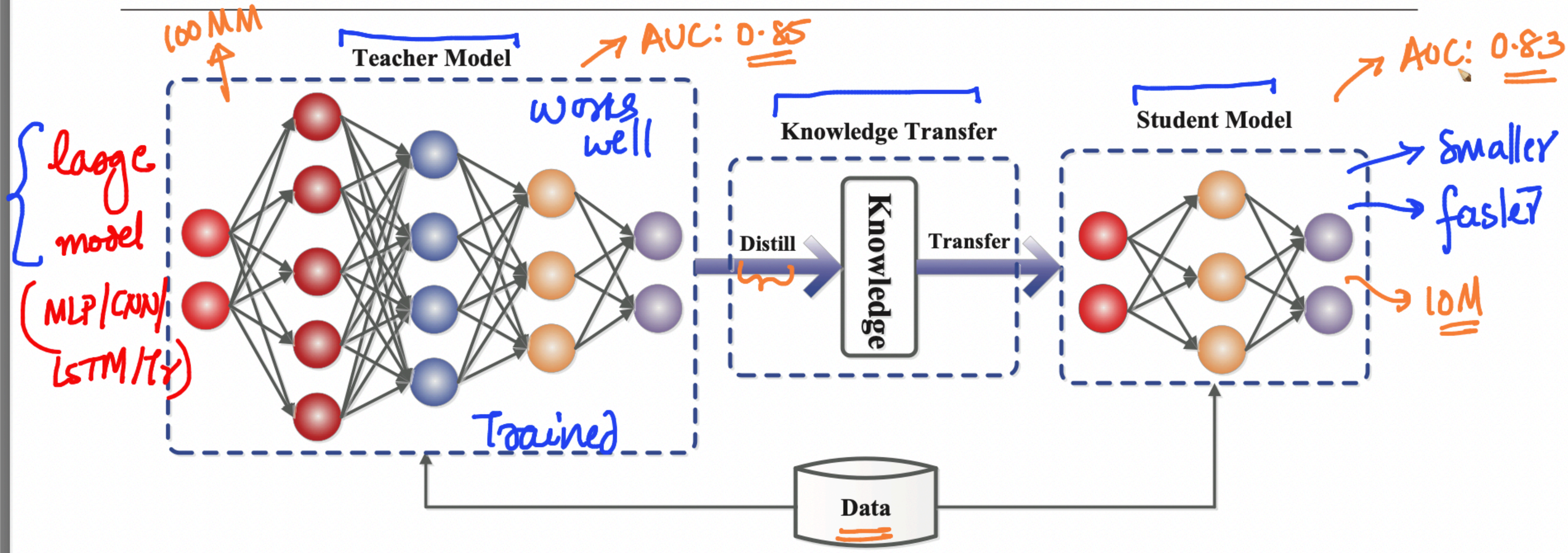


Fig. 1 The generic teacher-student framework for knowledge distillation.

2) model compression and acceleration techniques, in the following categories (Cheng et al., 2018).

- Parameter pruning and shari

due to the limited computational capacity and memory of the devices. To address this issue, Bucilua et al. (2006) first proposed model compression to transfer large model or an ensem-

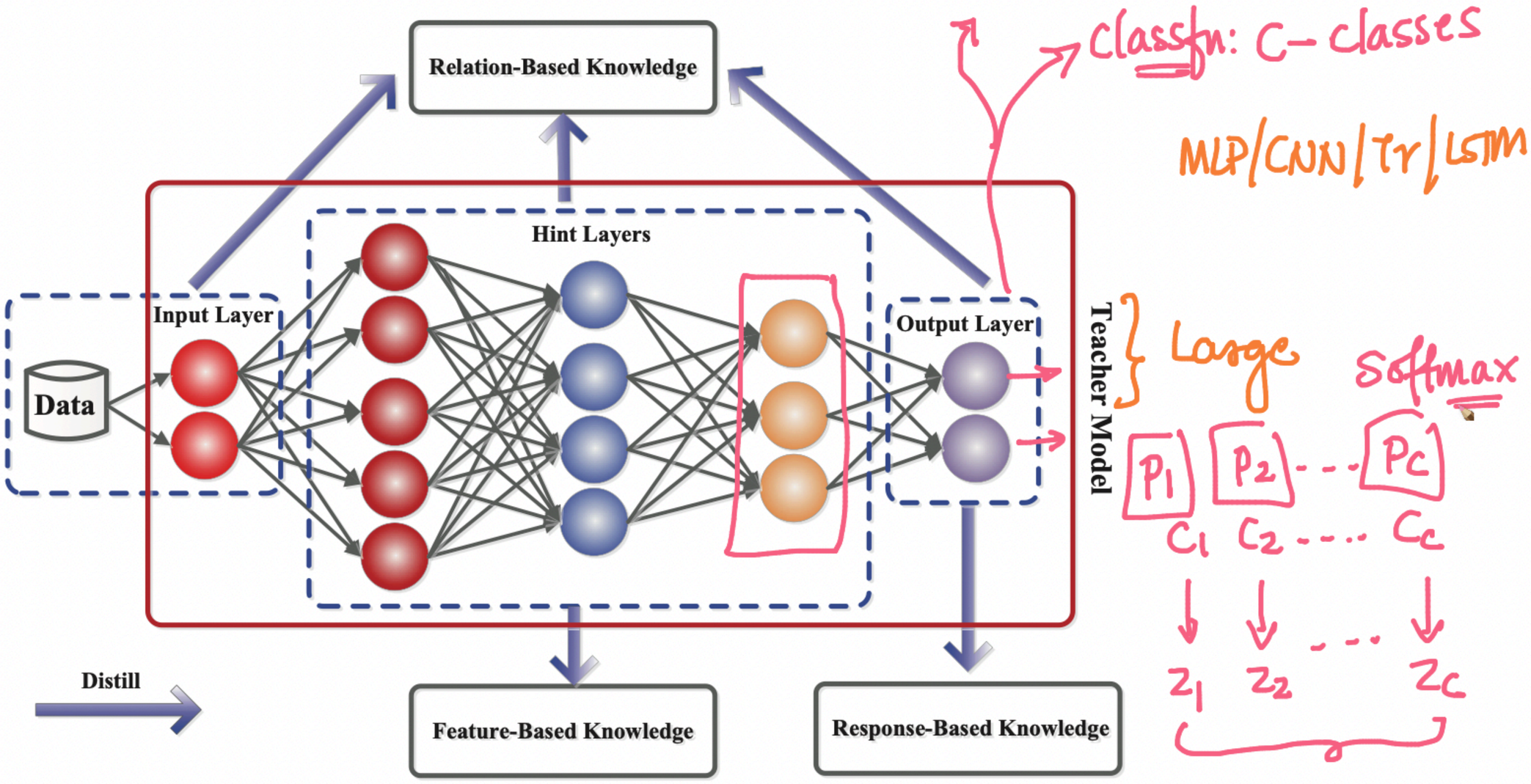


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network.

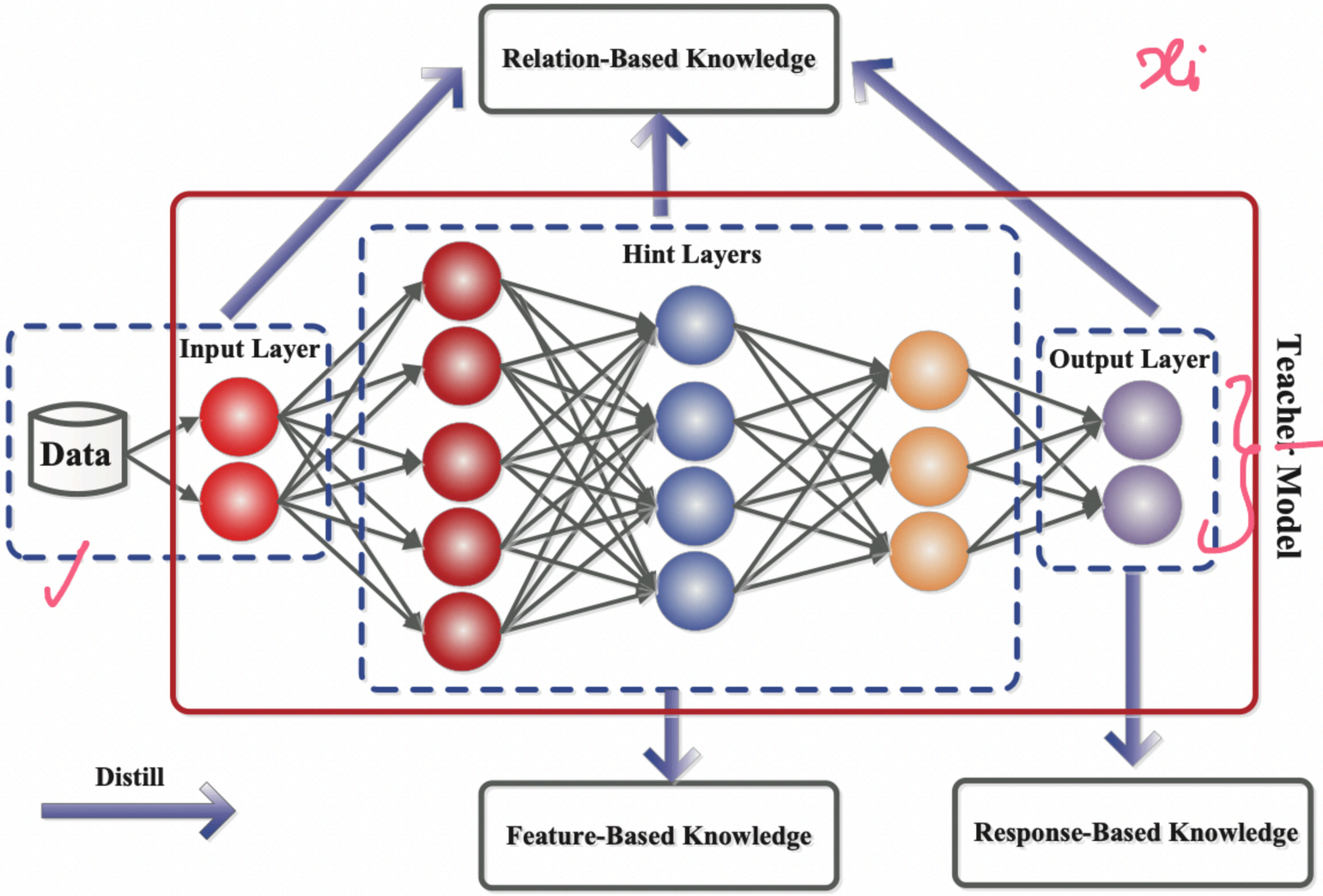


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network

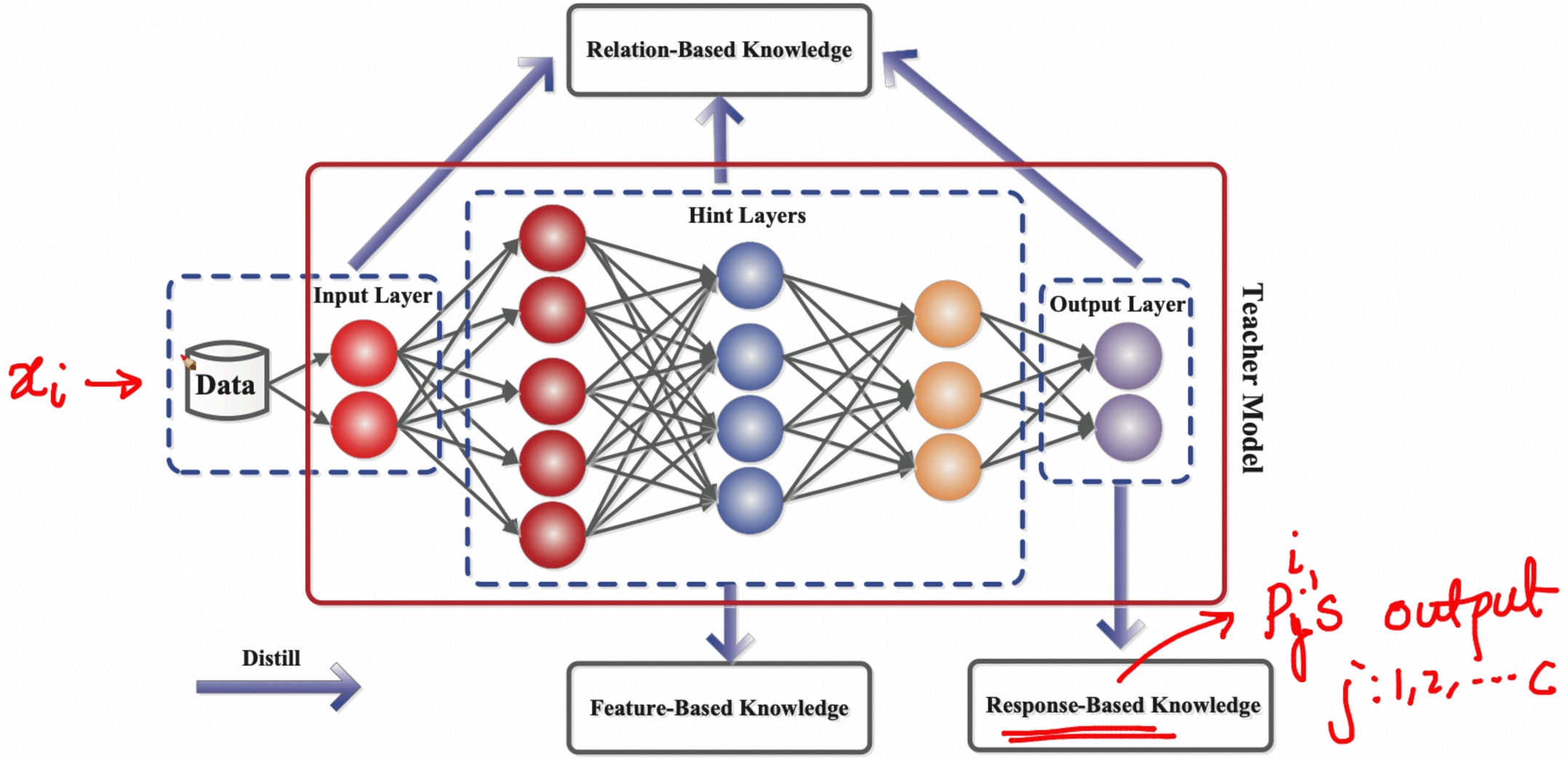
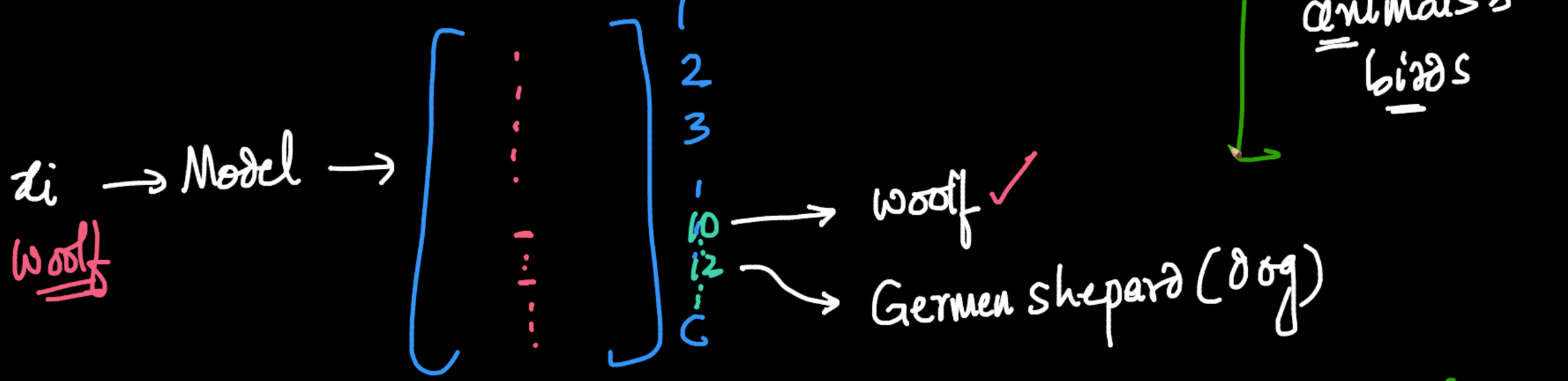


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network.

o/p probs/responses



implicit rich info in the o/p probs

(dark knowledge)

e.g: MNIST

4 9

3 8

5 6

x_i

dark-knowledge or implicit info

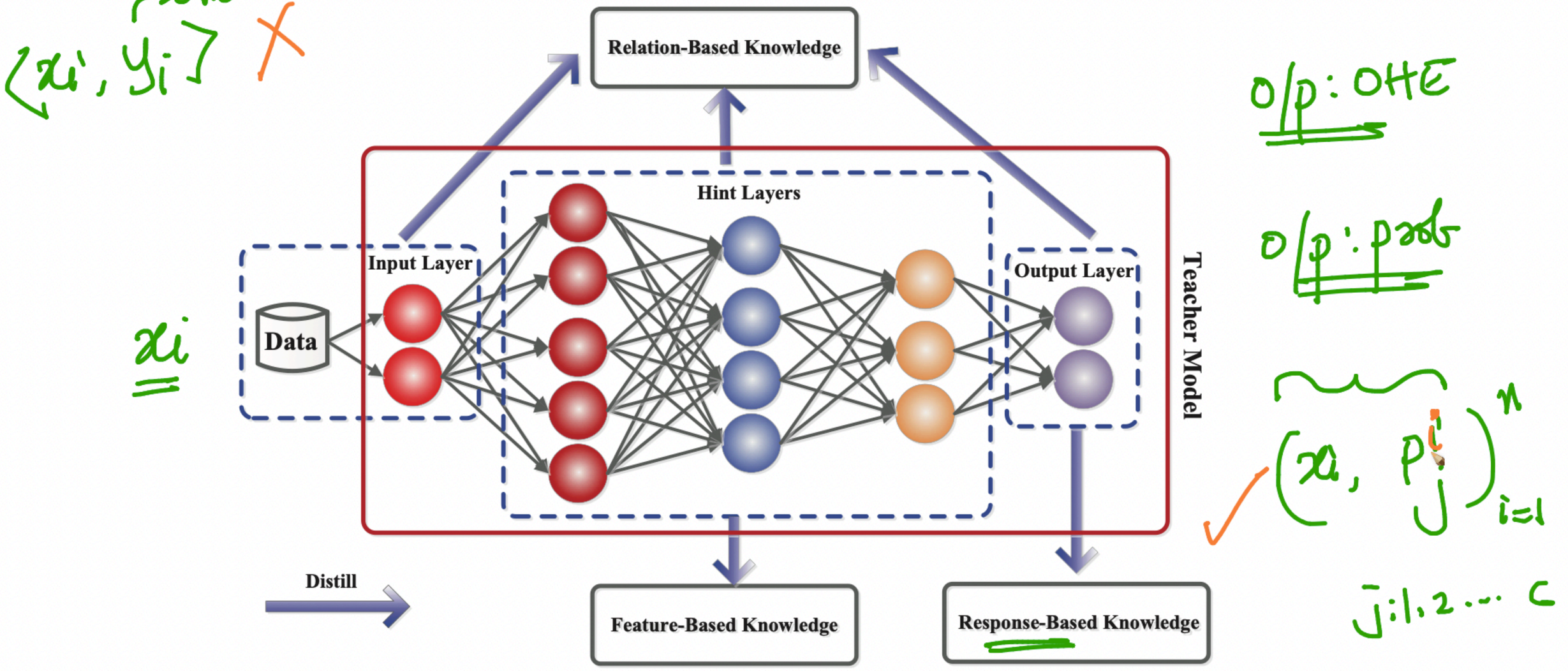


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network

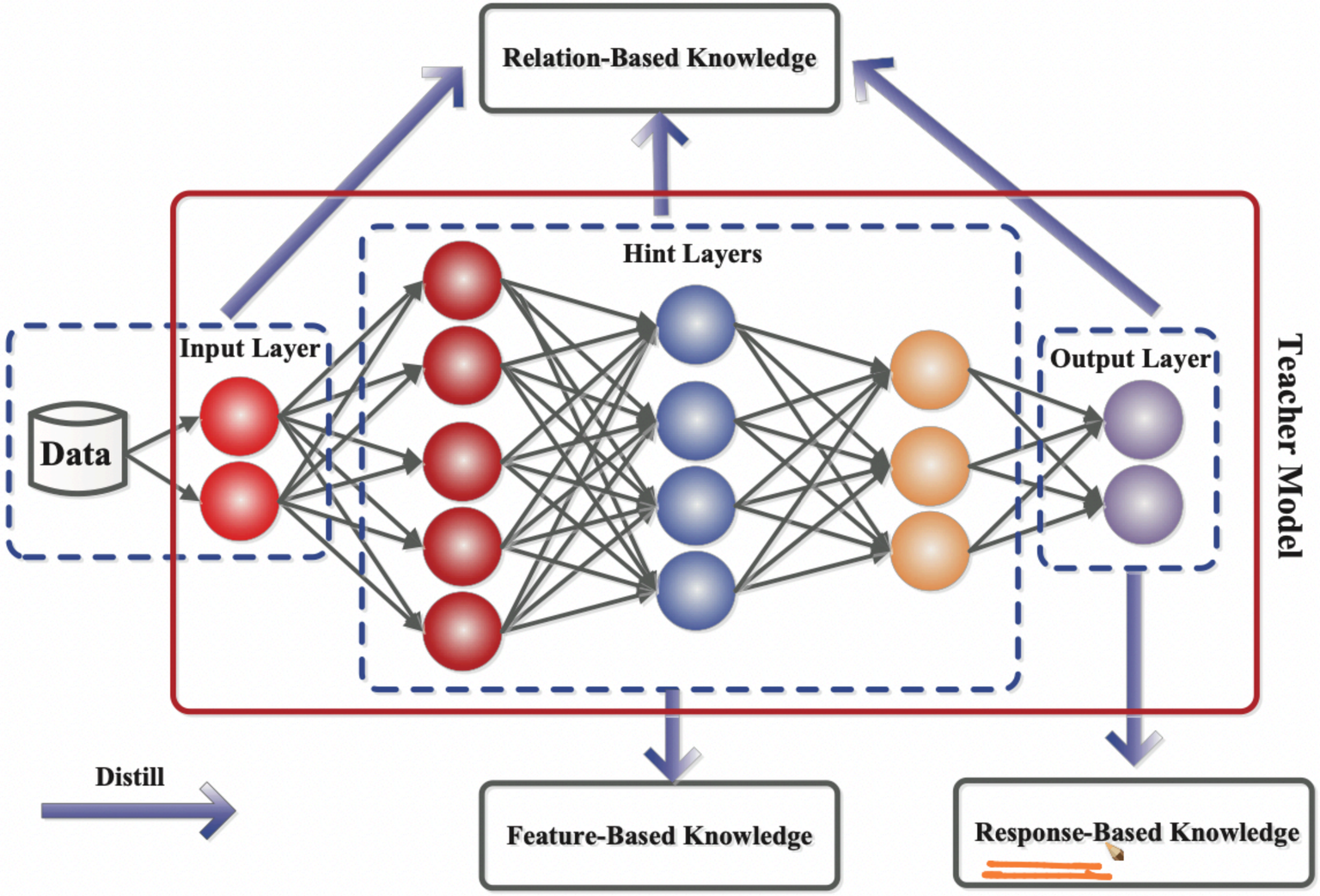


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network.

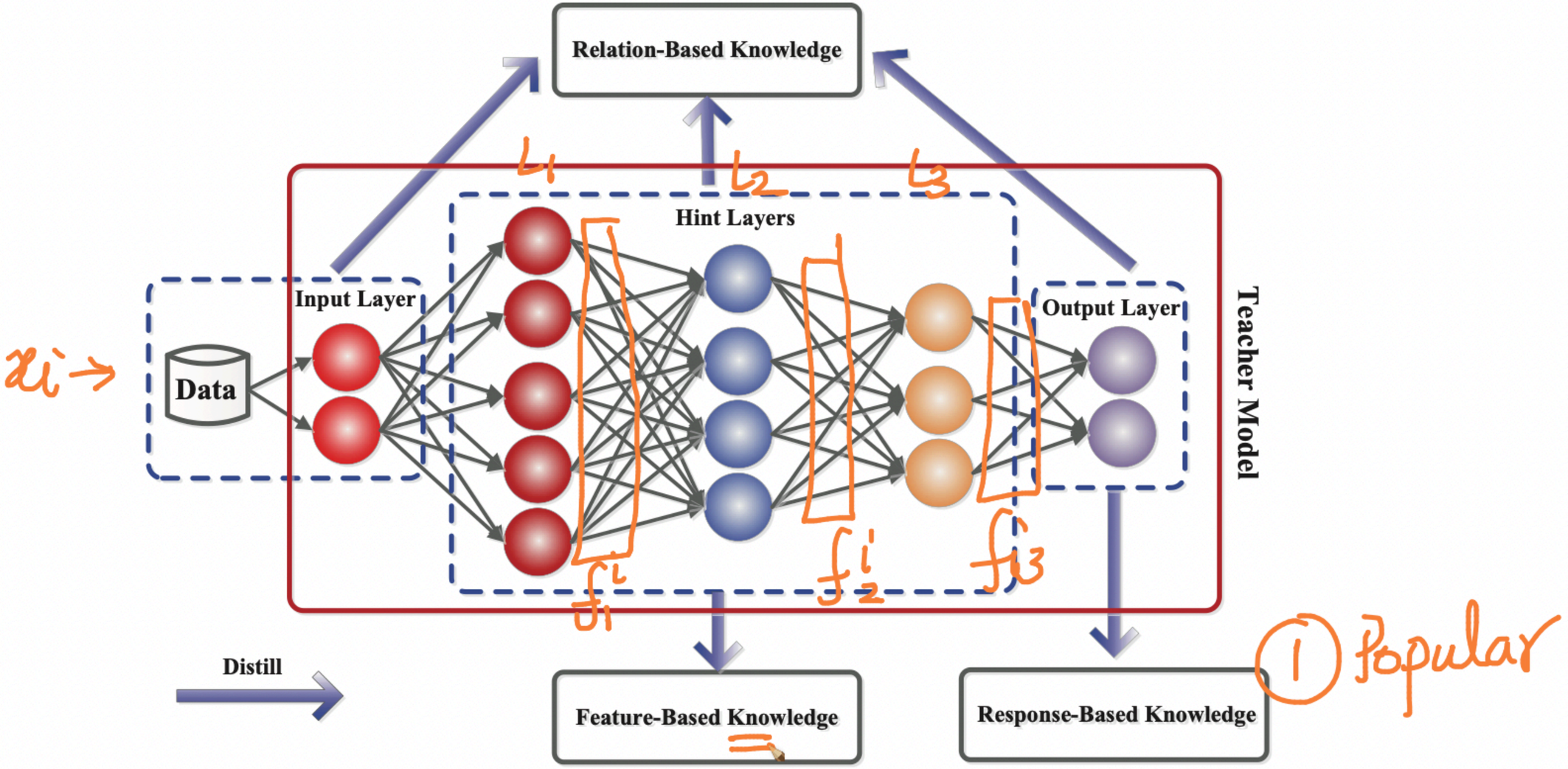


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network

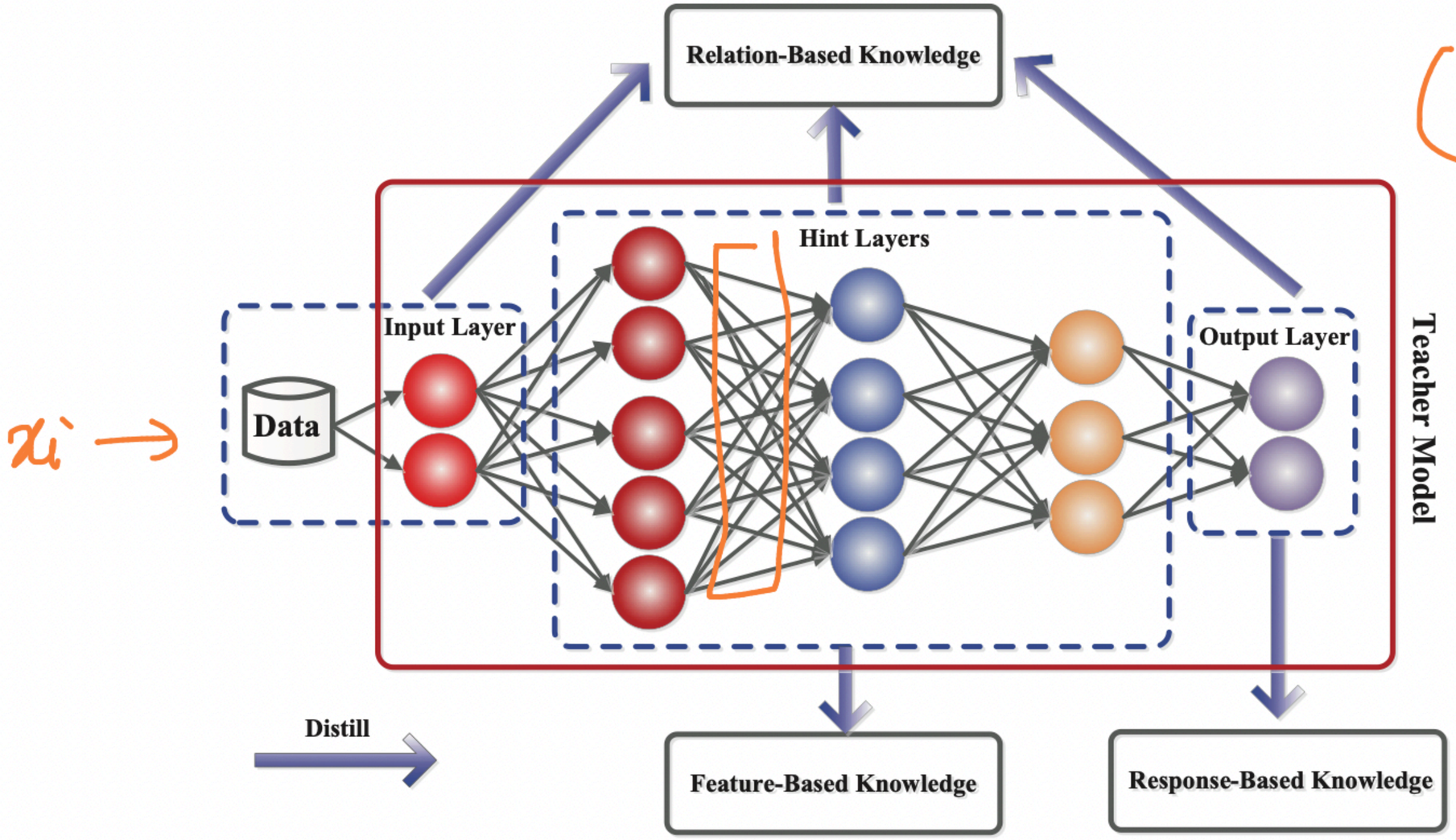


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network.

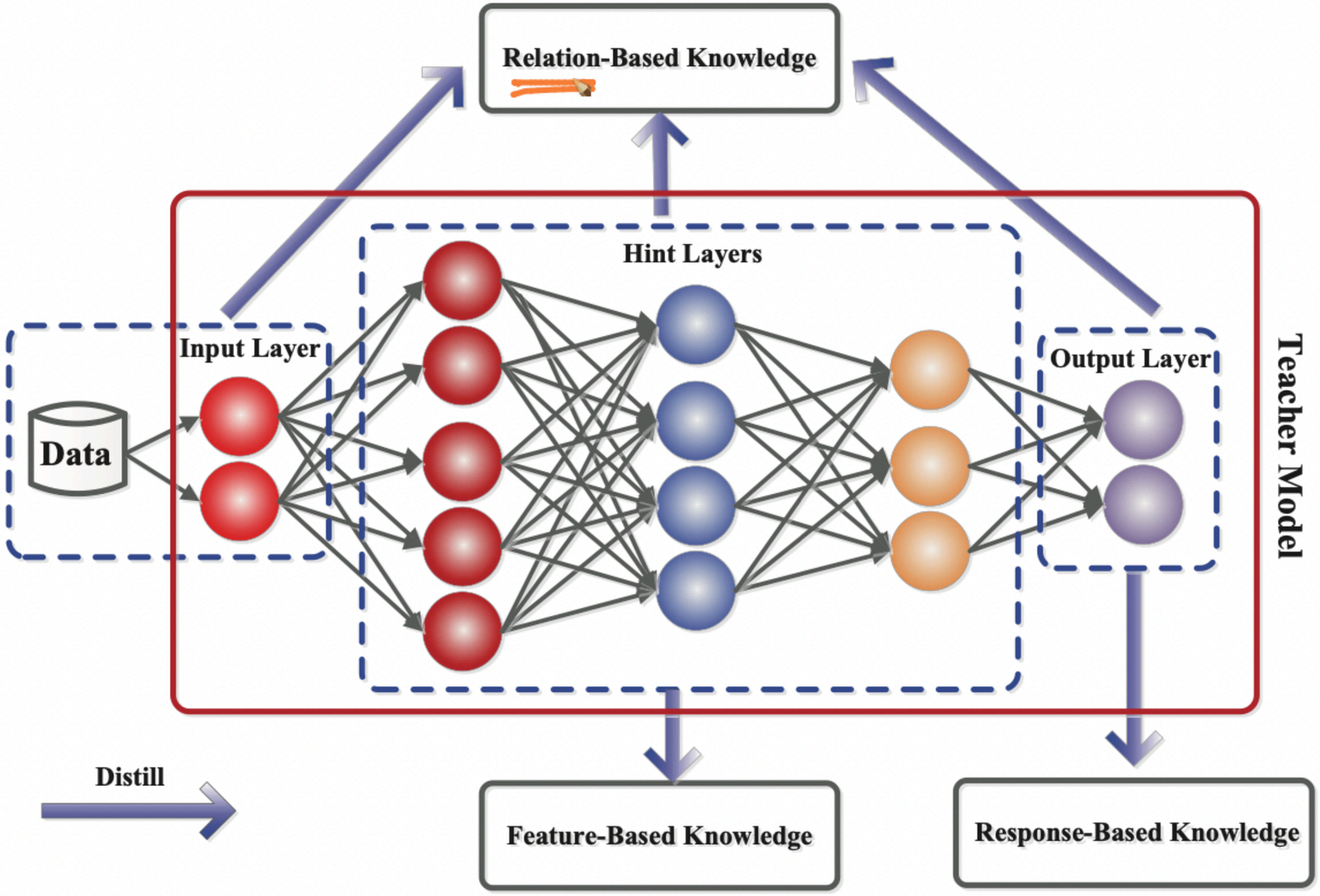


Fig. 3 The schematic illustrations of sources of response-based knowledge, feature-based knowledge and relation-based knowledge in a deep teacher network.

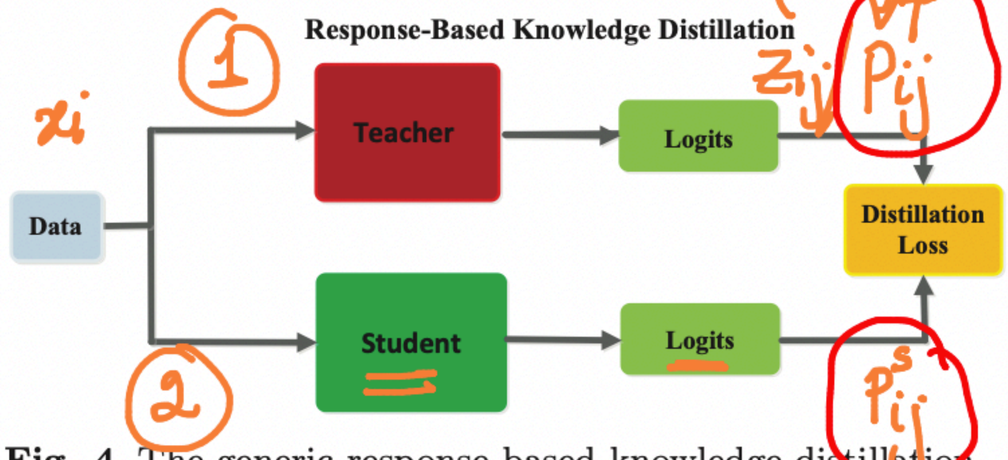


Fig. 4 The generic response-based knowledge distillation.

2.1 Response-Based Knowledge

Response-based knowledge usually refers to the neural response of the last output layer of the teacher model. The main idea is to directly mimic the final prediction of the teacher model. The response-based knowledge distillation is simple yet effective for model compression, and has been widely used in different tasks and applications. Given a vector of **logits** z as the outputs of the last fully connected laye

Generally, $\mathcal{L}_R(p(z_t, T), p(z_s, T))$ often employs Kullback-Leibler divergence loss. Clearly, optimizing Eq. (1) or (3) can make the logits z_s of student match the ones z_t of teacher. To easily understand the response-based knowledge distillation, the benchmark model of a vanilla knowledge distillation, which is the joint of the distillation and student losses, is given in Fig. 5. Note that the student loss is always defined as the cross-entropy loss $\mathcal{L}_{CE}(y, p(z_s, T = 1))$ between the ground truth label and the soft logits of the student model.

The idea of the response-based knowledge is straightforward and easy to understand, especially in the context of "dark knowledge". From another perspective, the effectiveness of the soft targets is analogous to label smoothing (Kim and Kim, 2017) or regularizers (Muller et al., 2019; Ding et al., 2019). However, the response-based knowledge usually relies on the output of the last layer, e.g., soft targets, and thus fails to address the intermediate-level supervision from the

rns out to be very important

Permanent link
Page information
Cite this page
Wikidata item

Print/export
Download as PDF
Printable version

Languages
Deutsch
Español
Français
한국어
Italiano
日本語
Português
Русский
中文
4 more
Edit links

Rate–distortion theory
Shannon's source coding theorem · Channel capacity · Noisy-channel coding theorem · Shannon–Hartley theorem
V • T • E

Definition [edit]

The cross-entropy of the distribution q relative to a distribution p over a given set is defined as follows:

$$H(p, q) = -\operatorname{E}_p[\log q],$$

where $E_p[\cdot]$ is the expected value operator with respect to the distribution p.

The definition may be formulated using the Kullback–Leibler divergence $D_{\mathrm{KL}}(p \parallel q)$, divergence of p from q (also known as the *relative entropy* of p with respect to q).

$$H(p, q) = H(p) + D_{\mathrm{KL}}(p \parallel q),$$

where $H(p)$ is the entropy of p.

For discrete probability distributions p and q with the same support $\mathcal{X}$ this means

$$H(p, q) = -\sum_{x \in \mathcal{X}} p(x) \log q(x) \quad \text{(Eq.1)}$$

The situation for continuous distributions is analogous. We have to assume that p and q are absolutely continuous with respect to some reference measure r (usually r is a Lebesgue measure on a Borel σ-algebra). Let P and Q be probability density functions of p and q with respect to r. Then

$$-\int_{\mathcal{X}} P(x) \log$$

16 / 17

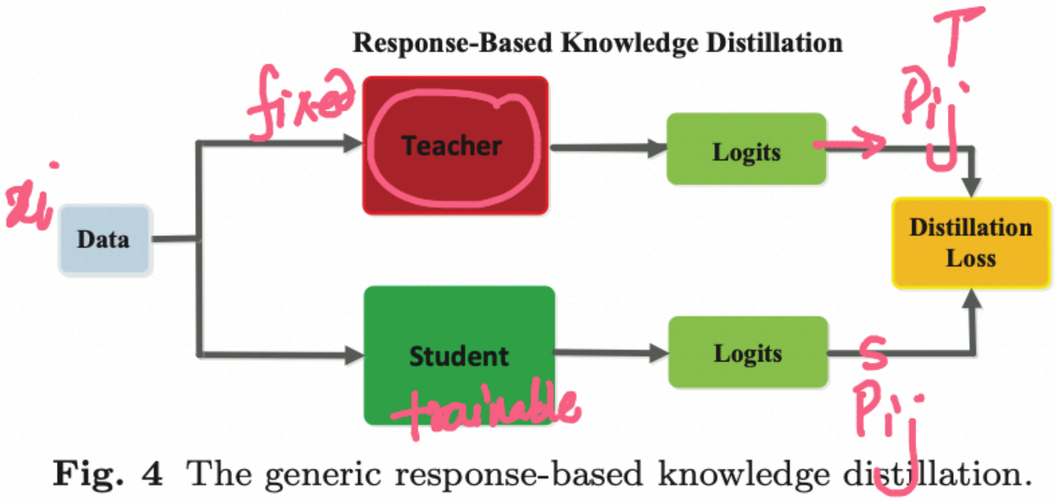


Fig. 4 The generic response-based knowledge distillation.

2.1 Response-Based Knowledge

Response-based knowledge usually refers to the neural response of the last output layer of the teacher model. The main idea is to directly mimic the final prediction of the teacher model. The response-based knowledge distillation is simple yet effective for model compression, and has been widely used in different tasks and applications. Given a vector of **logits** z as the outputs of the last fully connected laye

Generally, $\mathcal{L}_R(p(z_t, T), p(z_s, T))$ often employs Kullback-Leibler divergence loss. Clearly, optimizing Eq. (1) or (3) can make the logits z_s of student match the ones z_t of teacher. To easily understand the response-based knowledge distillation, the benchmark model of a vanilla knowledge distillation, which is the joint of the distillation and student losses, is given in Fig. 5. Note that the student loss is always defined as the cross-entropy loss $\mathcal{L}_{CE}(y, p(z_s, T = 1))$ between the ground truth label and the soft logits of the student model.

The idea of the response-based knowledge is straightforward and easy to understand, especially in the context of "dark knowledge". From another perspective, the effectiveness of the soft targets is analogous to label smoothing (Kim and Kim, 2017) or regularizers (Muller et al., 2019; Ding et al., 2019). However, the response-based knowledge usually relies on the output of the last layer, e.g., soft targets, and thus fails to address the intermediate-level supervision from the

rns out to be very important

✓ { log-loss = BCE

CE & KL

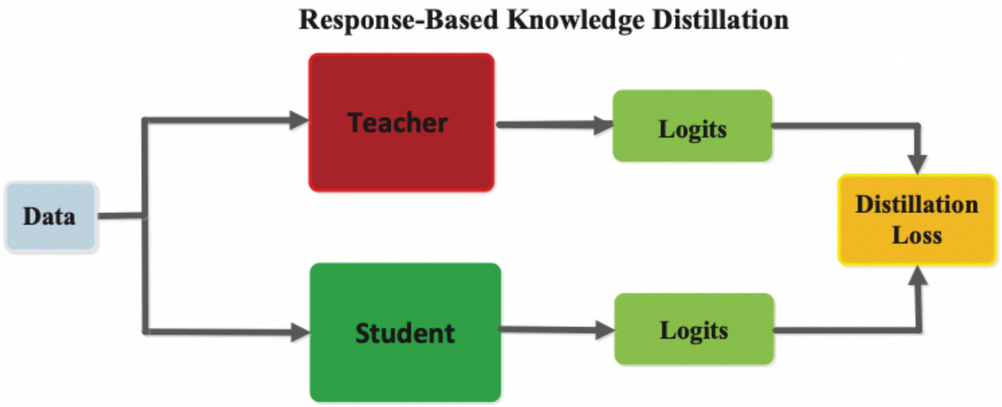


Fig. 4 The generic response-based knowledge distillation.

2.1 Response-Based Knowledge

Response-based knowledge usually refers to the neural response of the last output layer of the teacher model. The main idea is to directly mimic the final prediction of the teacher model. The response-based knowledge distillation is simple yet effective for model compression, and has been widely used in different tasks and applications. Given a vector of **logits** z as the outputs of the last fully connected laye

Generally, $\mathcal{L}_R(p(z_t, T), p(z_s, T))$ often employs Kullback-Leibler divergence loss. Clearly, optimizing Eq. (1) or (3) can make the logits z_s of student match the ones z_t of teacher. To easily understand the response-based knowledge distillation, the benchmark model of a vanilla knowledge distillation, which is the joint of the distillation and student losses, is given in Fig. 5. Note that the student loss is always defined as the cross-entropy loss $\mathcal{L}_{CE}(y, p(z_s, T = 1))$ between the ground truth label and the soft logits of the student model.

The idea of the response-based knowledge is straightforward and easy to understand, especially in the context of "dark knowledge". From another perspective, the effectiveness of the soft targets is analogous to label smoothing (Kim and Kim, 2017) or regularizers (Muller et al., 2019; Ding et al., 2019). However, the response-based knowledge usually relies on the output of the last layer, e.g., soft targets, and thus fails to address the intermediate-level supervision from the

rns out to be very important

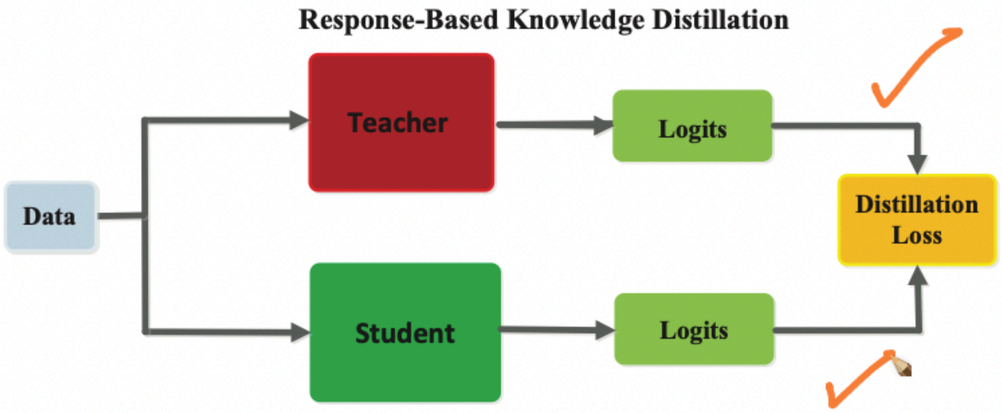


Fig. 4 The generic response-based knowledge distillation.

2.1 Response-Based Knowledge

Response-based knowledge usually refers to the neural response of the last output layer of the teacher model. The main idea is to directly mimic the final prediction of the teacher model. The response-based knowledge distillation is simple yet effective for model compression, and has been widely used in different tasks and applications. Given a vector of **logits** z as the outputs of the last fully connected laye

Generally, $\mathcal{L}_R(p(z_t, T), p(z_s, T))$ often employs Kullback-Leibler divergence loss. Clearly, optimizing Eq. (1) or (3) can make the logits z_s of student match the ones z_t of teacher. To easily understand the response-based knowledge distillation, the benchmark model of a vanilla knowledge distillation, which is the joint of the distillation and student losses, is given in Fig. 5. Note that the student loss is always defined as the cross-entropy loss $\mathcal{L}_{CE}(y, p(z_s, T = 1))$ between the ground truth label and the soft logits of the student model.

The idea of the response-based knowledge is straightforward and easy to understand, especially in the context of "dark knowledge". From another perspective, the effectiveness of the soft targets is analogous to label smoothing (Kim and Kim, 2017) or regularizers (Muller et al., 2019; Ding et al., 2019). However, the response-based knowledge usually relies on the output of the last layer, e.g., soft targets, and thus fails to address the intermediate-level supervision from the

rns out to be very important

most widely used

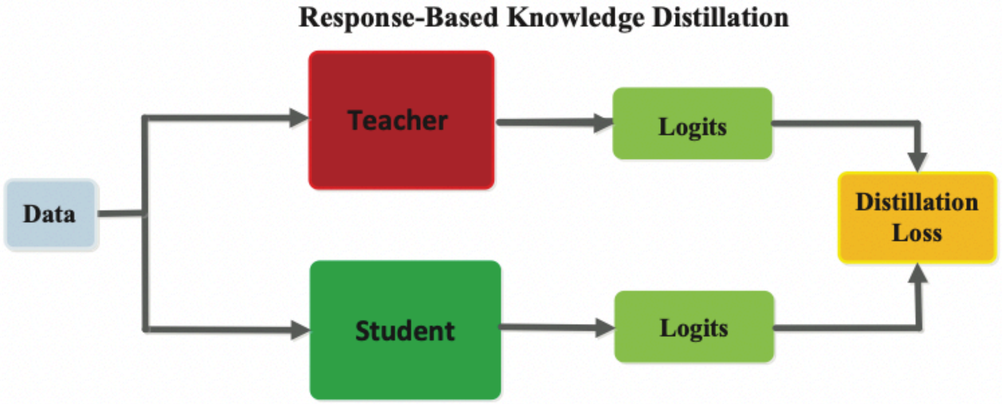


Fig. 4 The generic response-based knowledge distillation.

2.1 Response-Based Knowledge

Response-based knowledge usually refers to the neural response of the last output layer of the teacher model. The main idea is to directly mimic the final prediction of the teacher model. The response-based knowledge distillation is simple yet effect[illegible]wledge usually relies on the

Generally, $\mathcal{L}_R(p(z_t, T), p(z_s, T))$ often employs Kullback-Leibler divergence loss. Clearly, optimizing Eq. (1) or (3) can make the logits z_s of student match the ones z_t of teacher. To easily understand the response-based knowledge distillation, the benchmark model of a vanilla knowledge distillation, which is the joint of the distillation and student losses, is given in Fig. 5. Note that the student loss is always defined as the cross-entropy loss $\mathcal{L}_{CE}(y, p(z_s, T = 1))$ between the ground truth label and the soft logits of the student model.

The idea of the response-based knowledge is straightforward and easy to understand, especially in the context of "dark knowledge". From another perspective, the effectiveness of the soft targets is analogous to label smoothing (Kim and Kim, 2017) or regularizers (Müller et al., 2019; Ding et al., 2019). However,

EC-KD (Wang et al., 2020b)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
ALP-KD (Passban et al., 2021)	Attention-based layer projection	Hint layer	$\mathcal{L}_2(.)$
SemCKD (Chen et al., 2021)	Feature maps	Hint layer	$\mathcal{L}_2(.)$

Huang and Wang (2017) using neuron selectivity transfer. Passalis and Tefas (2018) transferred knowledge by matching the probability distribution in feature space. To make it easier to transfer the teacher knowledge, Kim et al. (2018) introduced so called "factors" as a more understandable form of intermediate representations. To reduce the performance gap between teacher and student, Jin et al. (2019) proposed route constrained hint learning, which supervises student by outputs of hint layers of teacher. Recently, Heo et al. (2019c) proposed to use the activation boundary of the hidden neurons for knowledge transfer. Interestingly, the parameter sharing of intermediate layers of the teacher model together with response-based knowledge is also used as the teacher knowledge (Zhou et al., 2018). To match the semantics between teacher and student, Chen et al. (2021) proposed cross-layer knowledge distillation, which adaptively assigns proper teacher layers for each student layer via attention allocation.

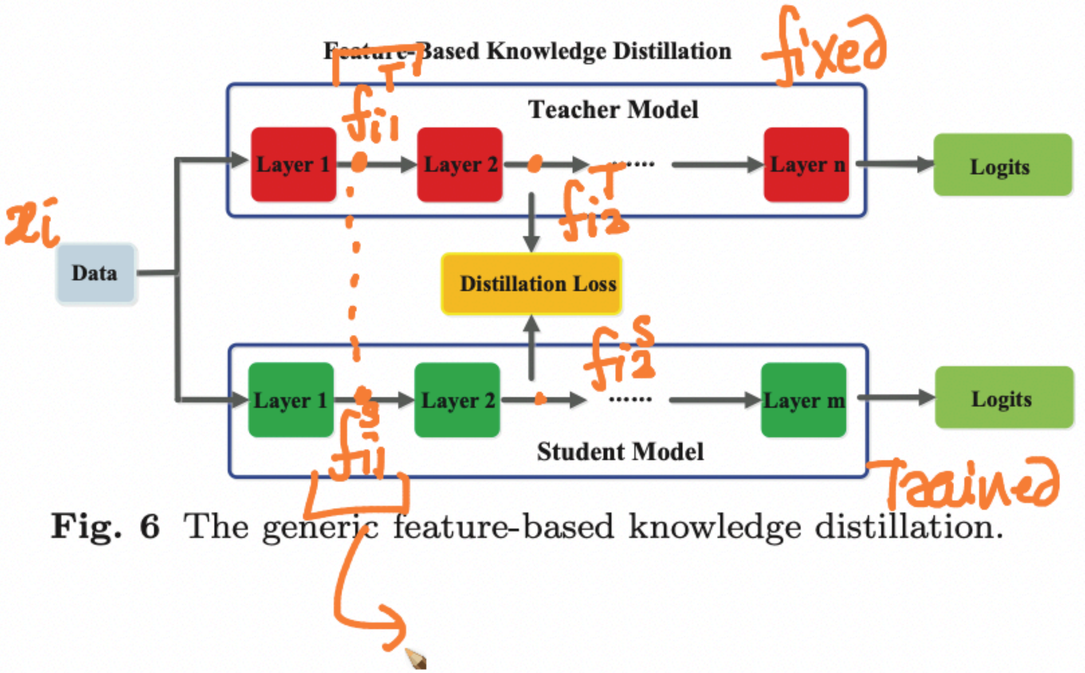


Fig. 6 The generic feature-based knowledge distillation.

where $f_t(x)$ and $f_s(x)$ are the feature maps of the intermediate layers of teacher and student models, respectively. The transformation functions, $\Phi_t(f_t(x))$ and $\Phi_s(f_s(x))$, are usually applied when the feature maps of teacher and student models are not in the same shape. $\mathcal{L}_F(.)$ indicates the similarity function used to match [illegible] and student models. A gen-

Generally, the distillation [illegible] is shown in Fig. 6. We also

using neuron selectivity trans-
018) transferred knowledge by
distribution in feature space.
ansfer the teacher knowledge,
duced so called "factors" as
form of intermediate repre-
he performance gap between
et al. (2019) proposed route
g, which supervises student by
f teacher. Recently, Heo et al.
the activation boundary of the
wledge transfer. Interestingly,
of intermediate layers of the
vith response-based knowledge
cher knowledge (Zhou et al.,
antics between teacher and stu

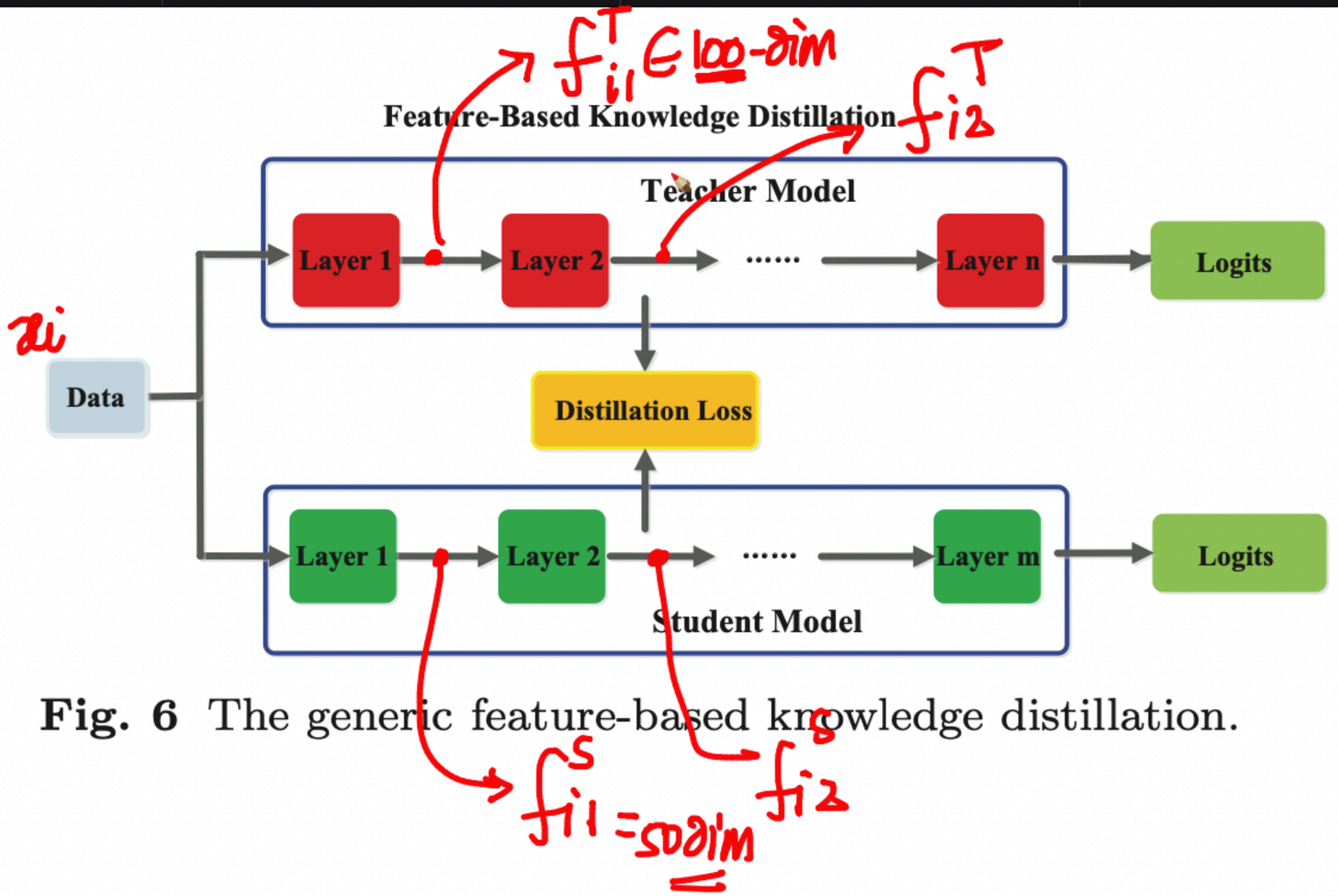
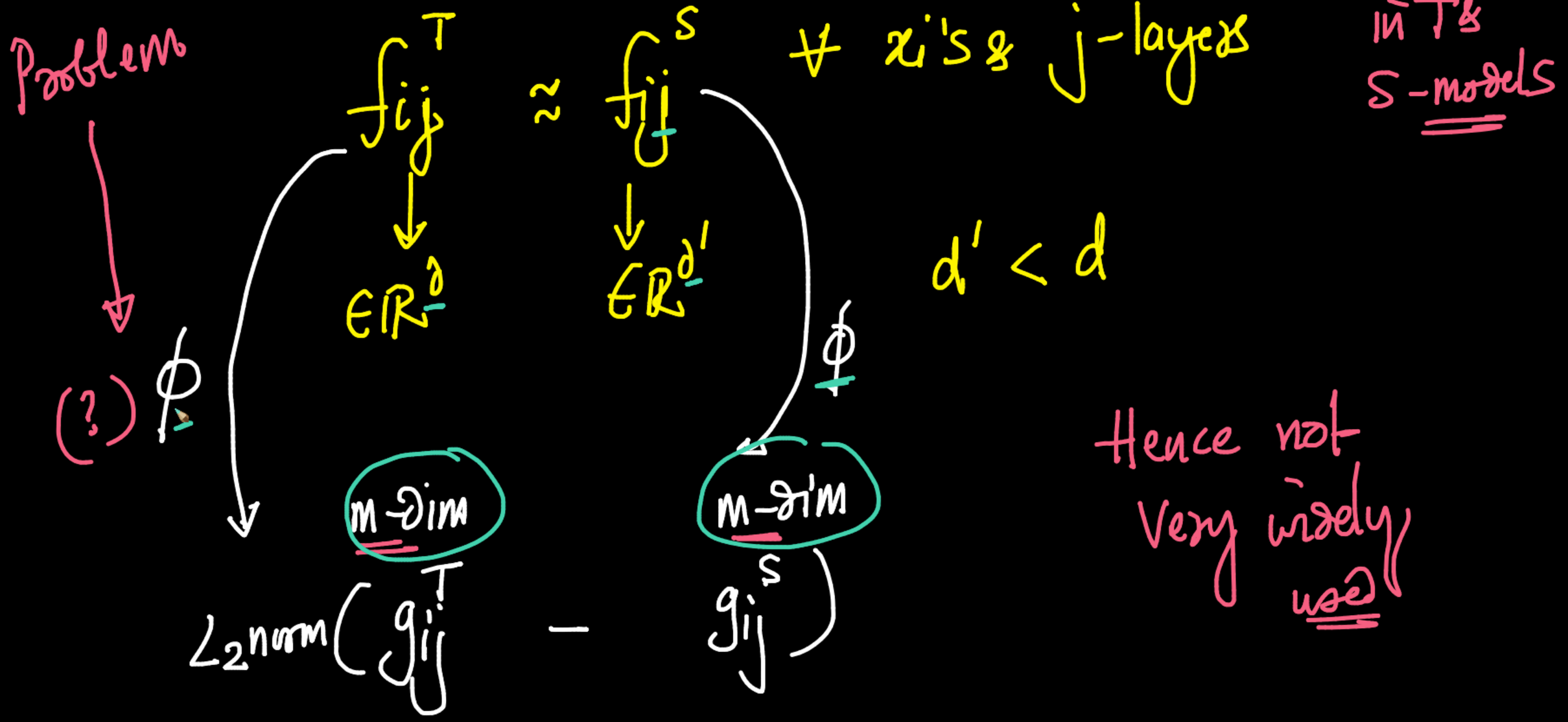


Fig. 6 The generic feature-based knowledge distillation.

where $f_t(x)$ and $f_s(x)$ are the feature maps of the intermediate layers of teacher and student models, respectively. The transformation functions, $\Phi_t(f_t(x))$ and

Feature-based KD:



using neuron selectivity trans-
018) transferred knowledge by
distribution in feature space.
ansfer the teacher knowledge,
duced so called "factors" as
form of intermediate repre-
he performance gap between
et al. (2019) proposed route
g, which supervises student by
f teacher. Recently, Heo et al.
the activation boundary of the
wledge transfer. Interestingly,
of intermediate layers of the
vith response-based knowledge
acher knowledge (Zhou et al.,
antics between teacher and stu

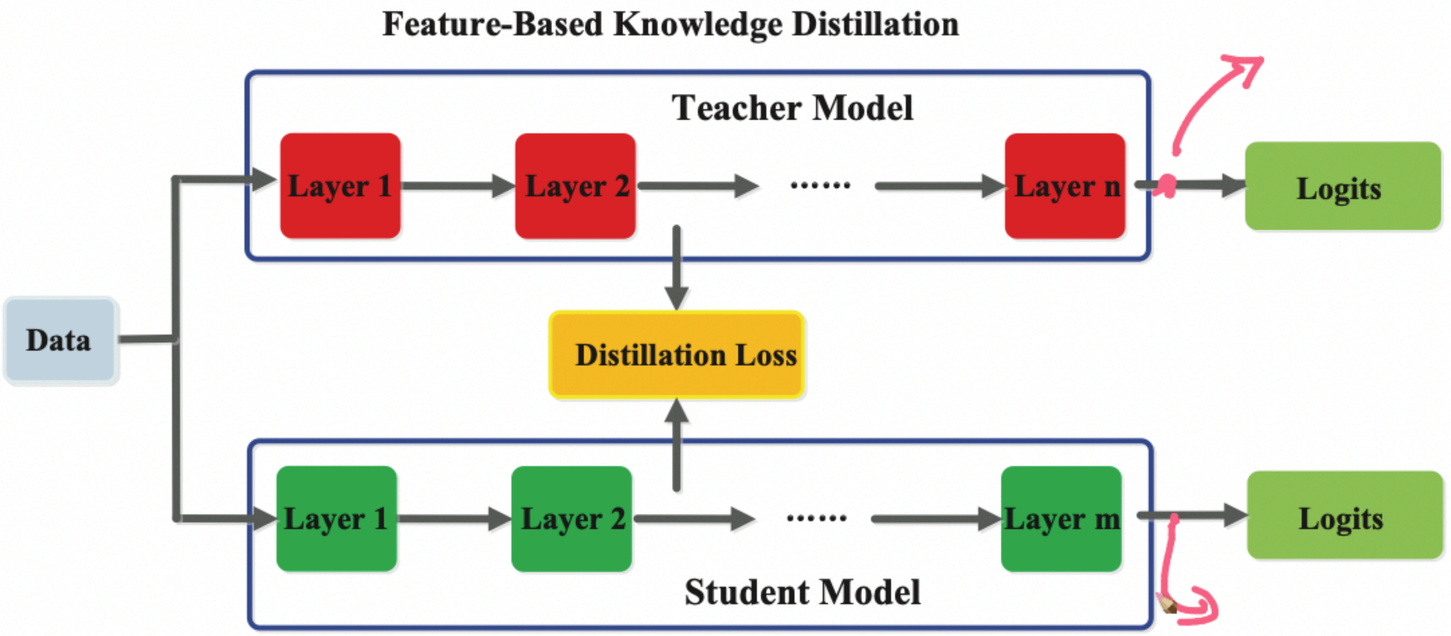


Fig. 6 The generic feature-based knowledge distillation.

where $f_t(x)$ and $f_s(x)$ are the feature maps of the intermediate layers of teacher and student models, respectively. The transformation functions, $\Phi_t(f_t(x))$ and

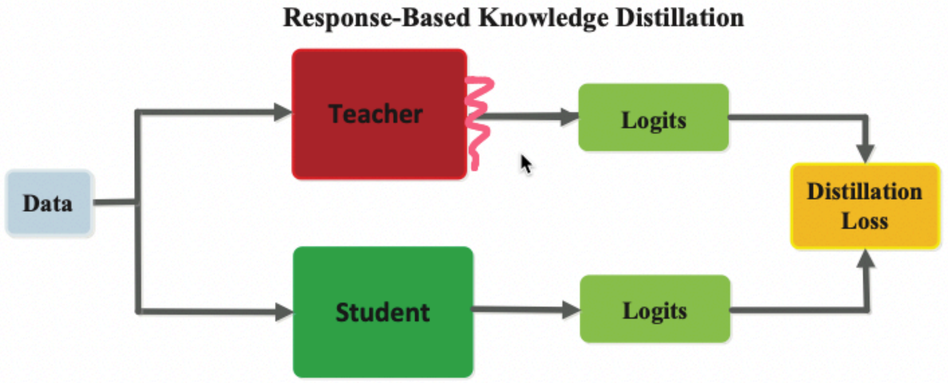


Fig. 4 The generic response-based knowledge distillation.

2.1 Response-Based Knowledge

Response-based knowledge usually refers to the neural response of the last output layer of the teacher model. The main idea is to directly mimic the final prediction of the teacher model. The response-based knowledge distillation is simple yet effective for model compression, and has been widely used in different tasks and applications. Given a vector

Generally, $\mathcal{L}_R(p(z_t, T), p(z_s, T))$ often employs Kullback-Leibler divergence loss. Clearly, optimizing Eq. (1) or (3) can make the logits z_s of student match the ones z_t of teacher. To easily understand the response-based knowledge distillation, the benchmark model of a vanilla knowledge distillation, which is the joint of the distillation and student losses, is given in Fig. 5. Note that the student loss is always defined as the cross-entropy loss $\mathcal{L}_{CE}(y, p(z_s, T = 1))$ between the ground truth label and the soft logits of the student model.

The idea of the response-based knowledge is straightforward and easy to understand, especially in the context of "dark knowledge". From another perspective, the effectiveness of the soft targets is analogous to label smoothing (Kim and Kim, 2017) or regularizers (Muller et al., 2019; Ding et al., 2019). However, the response-based knowledge usually relies on the output of the last layer, e.g., soft targets, and thus fails

en any two
twork. To
ne student
flow from
salis et al.,
tion-based
maps can

(5)

eacher and
e maps are
and from

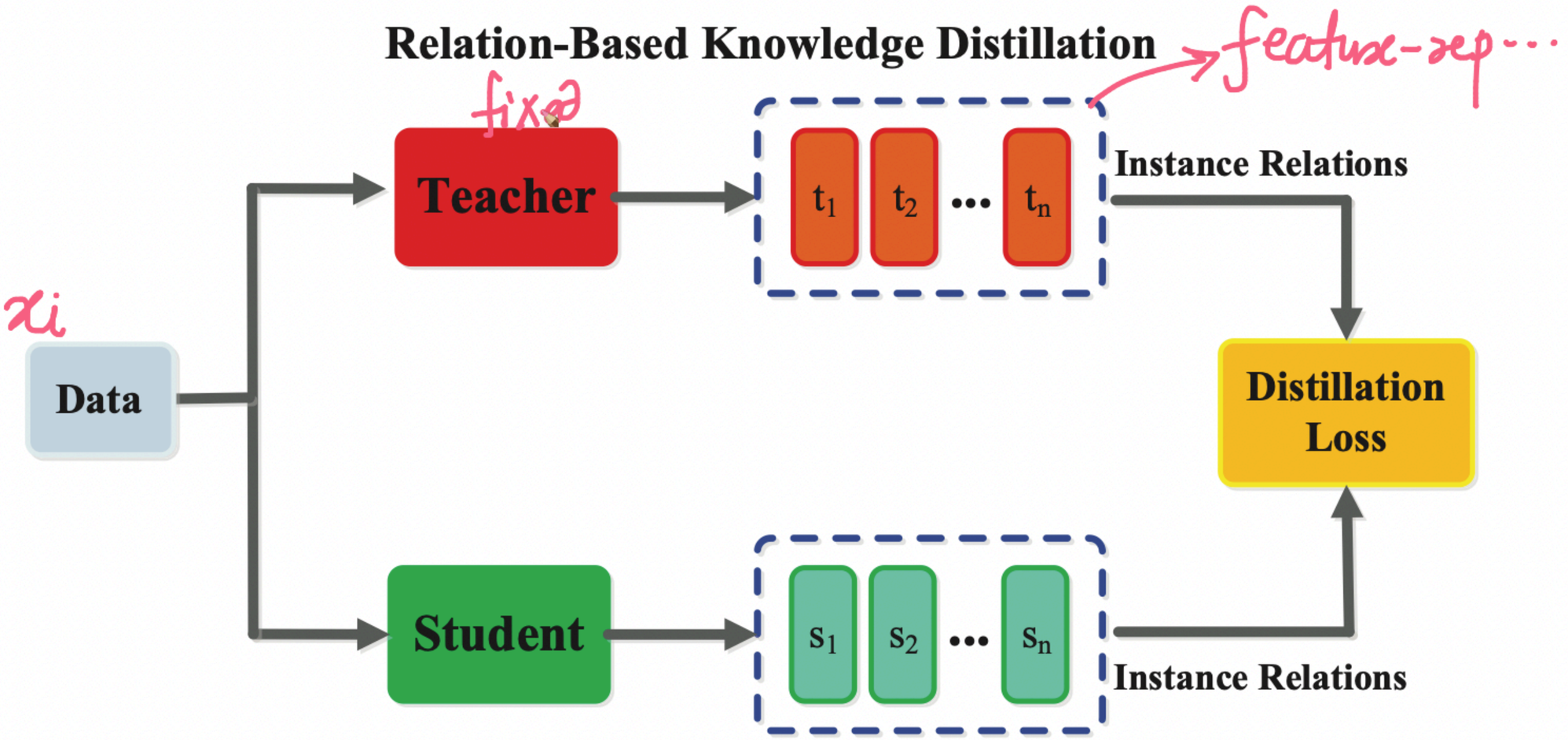


Fig. 7 The generic instance relation-based knowledge distillation.

and features of each teacher model as the nodes. Specifically, the importance and relationships of the different teachers are modeled by the logits and representation graphs before the knowledge transfer (Zhang and Peng, 2018). Multi-head graph-based knowledge distillation was proposed by Lee and Song (2019). The graph knowledge is the intra-data relations between any two feature maps via multi-head attention network. To explore the pairwise hint information, the student model also mimics the mutual information flow from pairs of hint layers of the teacher model (Passalis et al., 2020b). In general, the distillation loss of relation-based knowledge based on the relations of feature maps can be formulated as

$$L_{RelD}(f_t, f_s) = \mathcal{L}_{R^1}\left(\Psi_t(\hat{f}_t, \check{f}_t), \Psi_s(\hat{f}_s, \check{f}_s)\right), \quad (5)$$

where f_t and f_s are the feature maps of teacher and student models, respectively. Pairs of feature maps are chosen from the teacher model, $\hat{f}_t$ and $\check{f}_t$, and from the student model, $\hat{f}_s$ and $\check{f}_s$. $\Psi_t(.)$ and $\Psi_s(.)$ are the similarity functions for pairs of feature maps from [illegible] knowledge based on the instance relations can be the teacher and student models.

contains both the instance-level information and the correlations between instances. Using the correlation congruence for distillation, the student network can learn the correlation between instances.

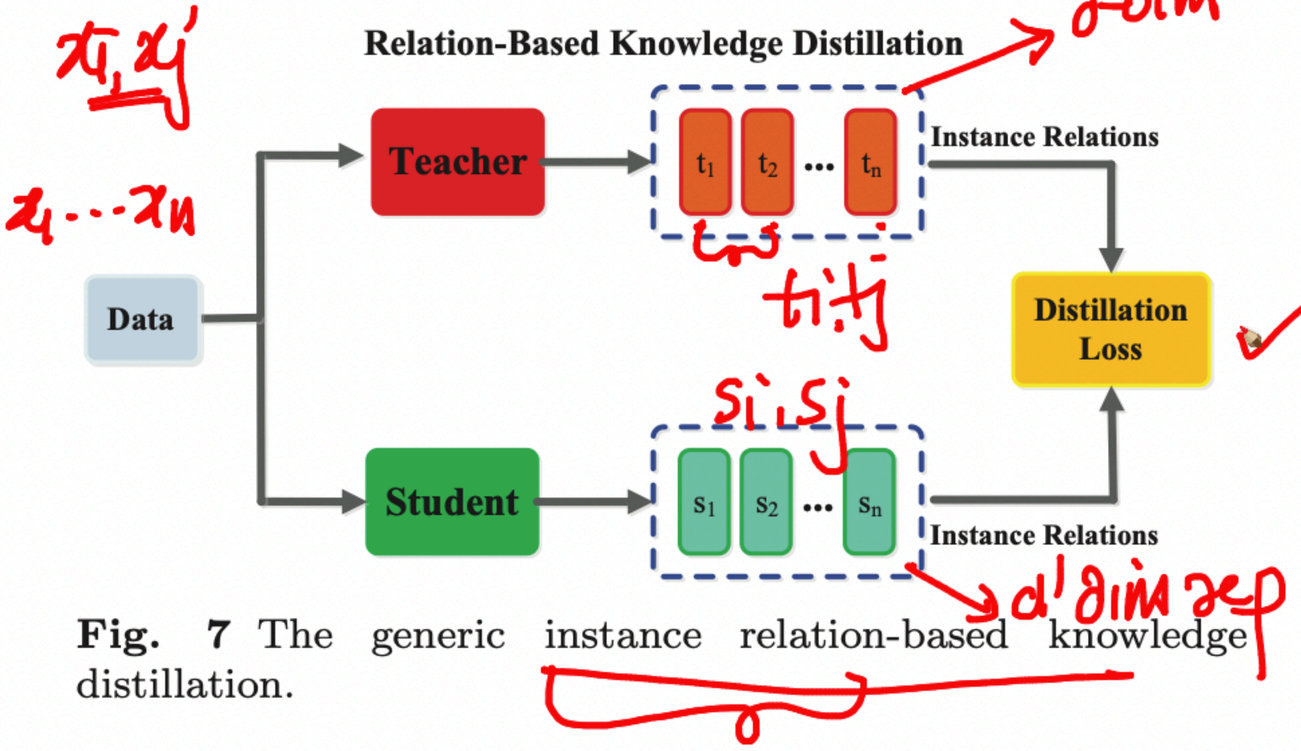


Fig. 7 The generic instance relation-based knowledge distillation.

As described above, the distillation loss of relation-

and features of each teacher model as the nodes. Specifically, the importance and relationships of the different teachers are modeled by the logits and representation graphs before the knowledge transfer (Zhang and Peng, 2018). Multi-head graph-based knowledge distillation was proposed by Lee and Song (2019). The graph knowledge is the intra-data relations between any two feature maps via multi-head attention network. To explore the pairwise hint information, the student model also mimics the mutual information flow from pairs of hint layers of the teacher model (Passalis et al., 2020b). In general, the distillation loss of relation-based knowledge based on the relations of feature maps can be formulated as

$$L_{RelD}(f_t, f_s) = \mathcal{L}_{R^1}\left(\Psi_t(\hat{f}_t, \check{f}_t), \Psi_s(\hat{f}_s, \check{f}_s)\right), \quad (5)$$

where f_t and f_s are the feature maps of teacher and student models, respectively. Pairs of feature maps are chosen from the teacher model, $\hat{f}_t$ and $\check{f}_t$, and from the student model, $\hat{f}_s$ and $\check{f}_s$. $\Psi_t(.)$ and $\Psi_s(.)$ are the similarity functions for pairs of feature maps from the teacher and student models

contains both the instance-level information and the correlations between instances. Using the correlation congruence for distillation, the student network can learn the correlation between instances.

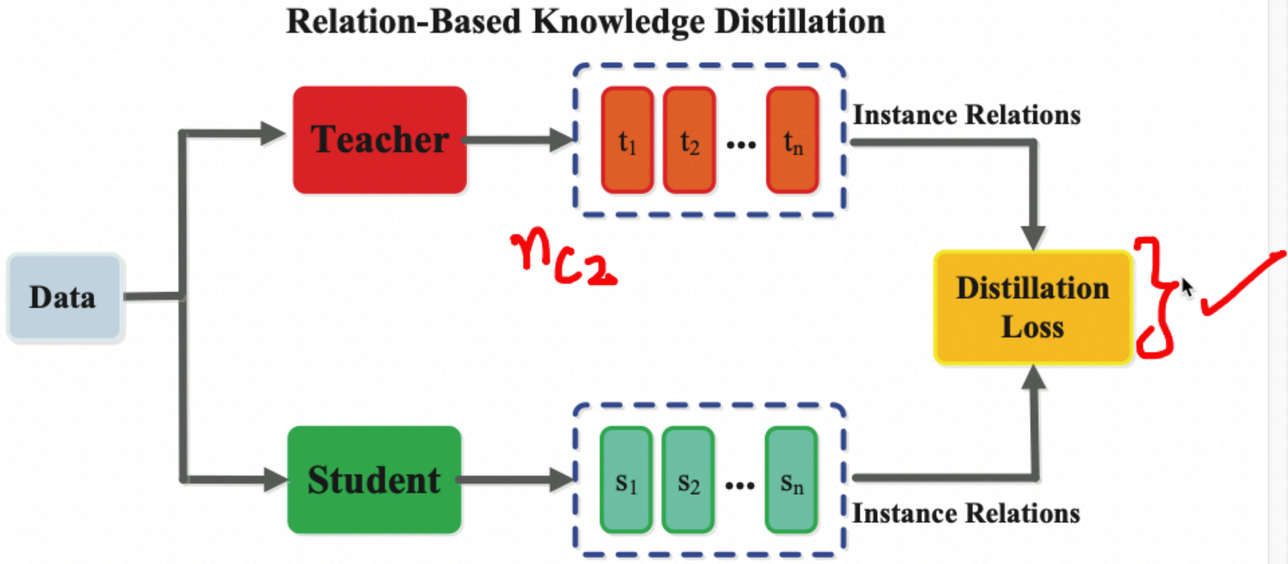


Fig. 7 The generic instance relation-based knowledge distillation.

As described above, the distillation loss of relation-based knowledge based on the instance relations can be

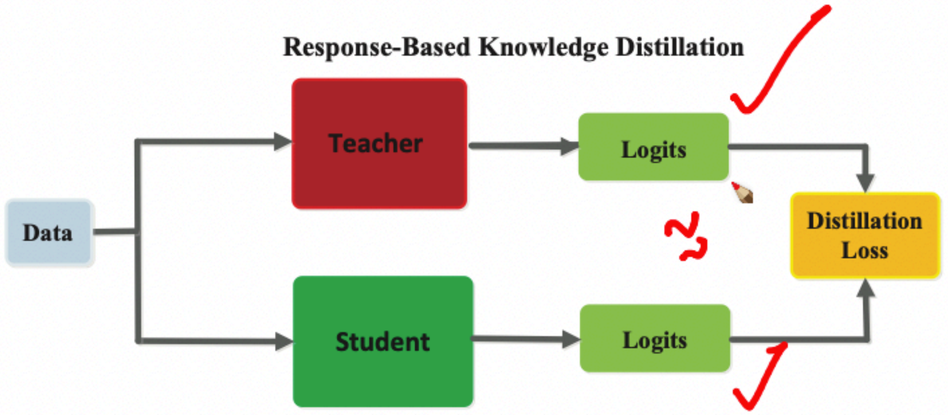


Fig. 4 The generic response-based knowledge distillation.

2.1 Response-Based Knowledge

Response-based knowledge usually refers to the neural response of the last output layer of the teacher model. The main idea is to directly mimic the final prediction of the teacher model. The r

Generally, $\mathcal{L}_R(p(z_t, T), p(z_s, T))$ often employs Kullback-Leibler divergence loss. Clearly, optimizing Eq. (1) or (3) can make the logits z_s of student match the ones z_t of teacher. To easily understand the response-based knowledge distillation, the benchmark model of a vanilla knowledge distillation, which is the joint of the distillation and student losses, is given in Fig. 5. Note that the student loss is always defined as the cross-entropy loss $\mathcal{L}_{CE}(y, p(z_s, T = 1))$ between the ground truth label and the soft logits of the student model.

The idea of the response-based knowledge is straightforward and easy to understand, especially in the context of "dark knowledge". From another perspective, the effectiveness of the soft targets is analogous to label smoothing (Kim and Kim, 2017) or regulariz- Ding et al., 2019). However,

ically, the importance and relationships of the different teachers are modeled by the logits and representation graphs before the knowledge transfer (Zhang and Peng, 2018). Multi-head graph-based knowledge distillation was proposed by Lee and Song (2019). The graph knowledge is the intra-data relations between any two feature maps via multi-head attention network. To explore the pairwise hint information, the student model also mimics the mutual information flow from pairs of hint layers of the teacher model (Passalis et al., 2020b). In general, the distillation loss of relation-based knowledge based on the relations of feature maps can be formulated as

$$L_{RelD}(f_t, f_s) = \mathcal{L}_{R^1}\left(\Psi_t(\hat{f}_t, \check{f}_t), \Psi_s(\hat{f}_s, \check{f}_s)\right) , \quad (5)$$

where f_t and f_s are the feature maps of teacher and student models, respectively. Pairs of feature maps are chosen from the teacher model, $\hat{f}_t$ and $\check{f}_t$, and from the student model, $\hat{f}_s$ and $\check{f}_s$. $\Psi_t(.)$ and $\Psi_s(.)$ are the similarity functions for pairs of feature maps from the teacher and student models. $\mathcal{L}_{R^1}(.)$ indicates the correlation function between the teacher and student feature maps.

contains both the instance-level information and the correlations between instances. Using the correlation congruence for distillation, the student network can learn the correlation between instances.

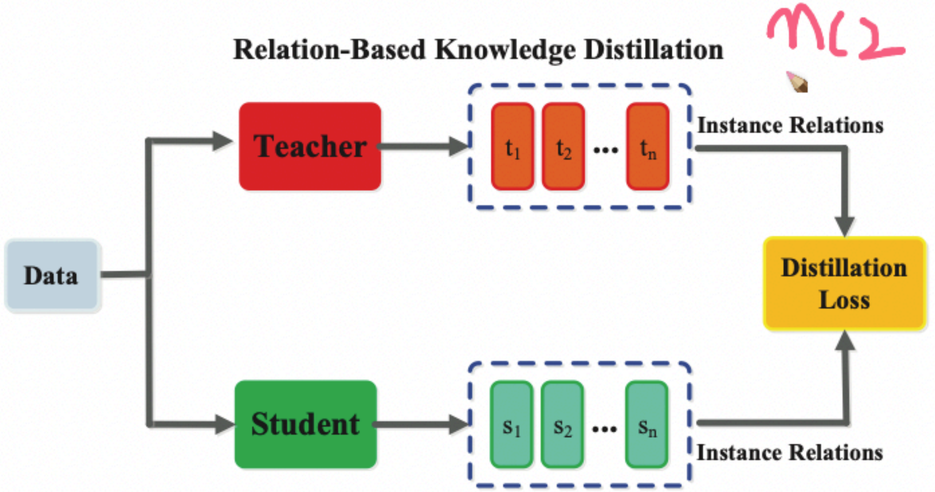


Fig. 7 The generic instance relation-based knowledge distillation.

As described above, the distillation loss of relation-based knowledge based on the instance relations can be formulated as

$$L_{RelD}(F_t, F_s) = \mathcal{L}_{R^2}\left(\psi_t(t_i, t_j), \psi_s(s_i, s_j)\right) , \quad (6)$$

Transfer learning

Imagenet → Model → Mc.classfn

‖ 22 ‖

Xray images → Model → X-ray classfn

KD

$\mathcal{D}$ → M_T^{Large} → MC class..

‖ ‖ ‖

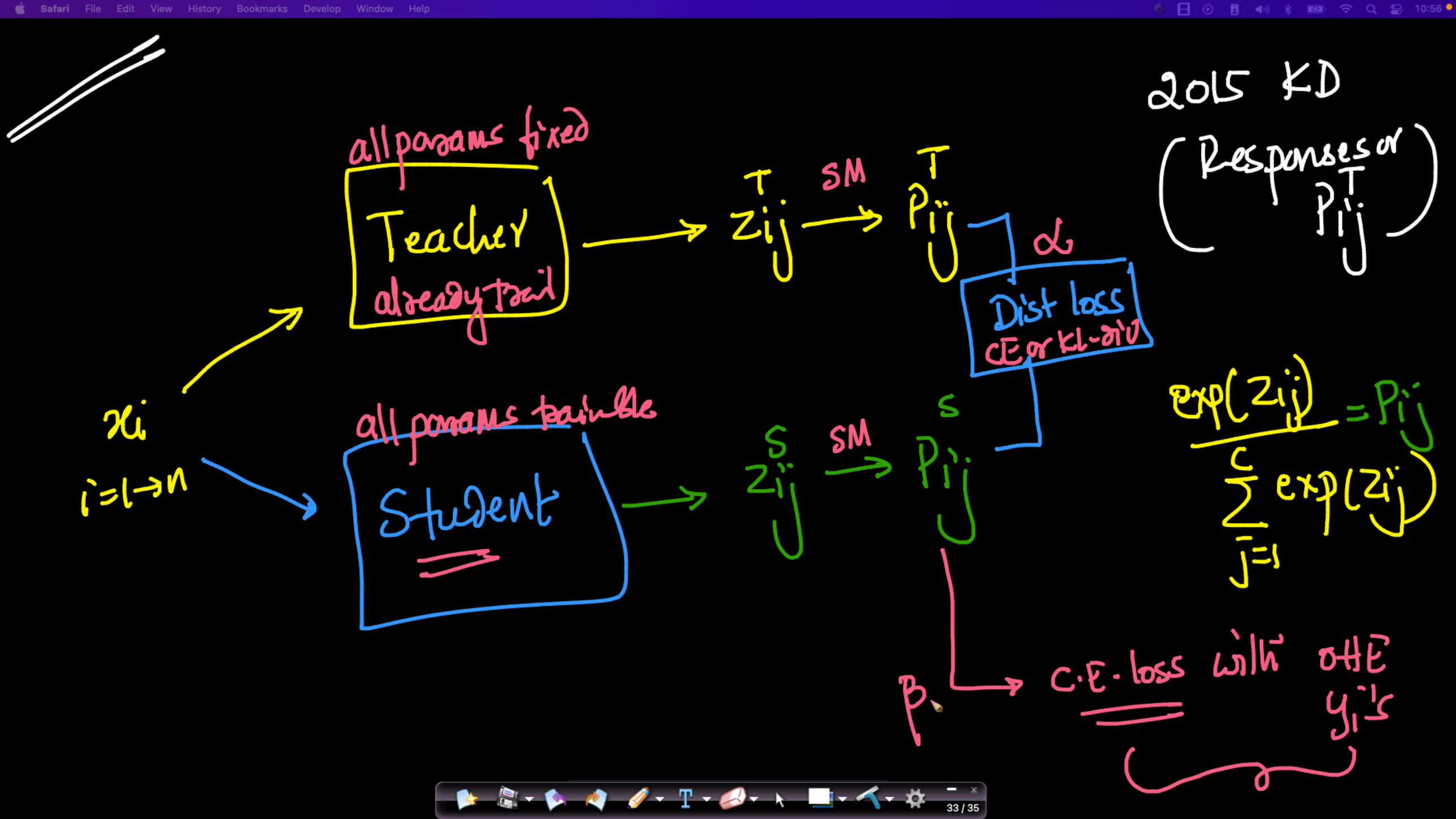
$\mathcal{D}$ → $M_S^{Smaller}$ → MC-Classfn

Respone-based KD

↳ Most popular

↓

2015 Hinton, ..., Dean...



correct answer turns out to be helpful.

2 Distillation

Neural networks typically produce class probabilities by using a "softmax" output layer that converts the logit, z_i, computed for each class into a probability, q_i, by comparing z_i with the other logits.

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)} \quad (1)$$

Temperature

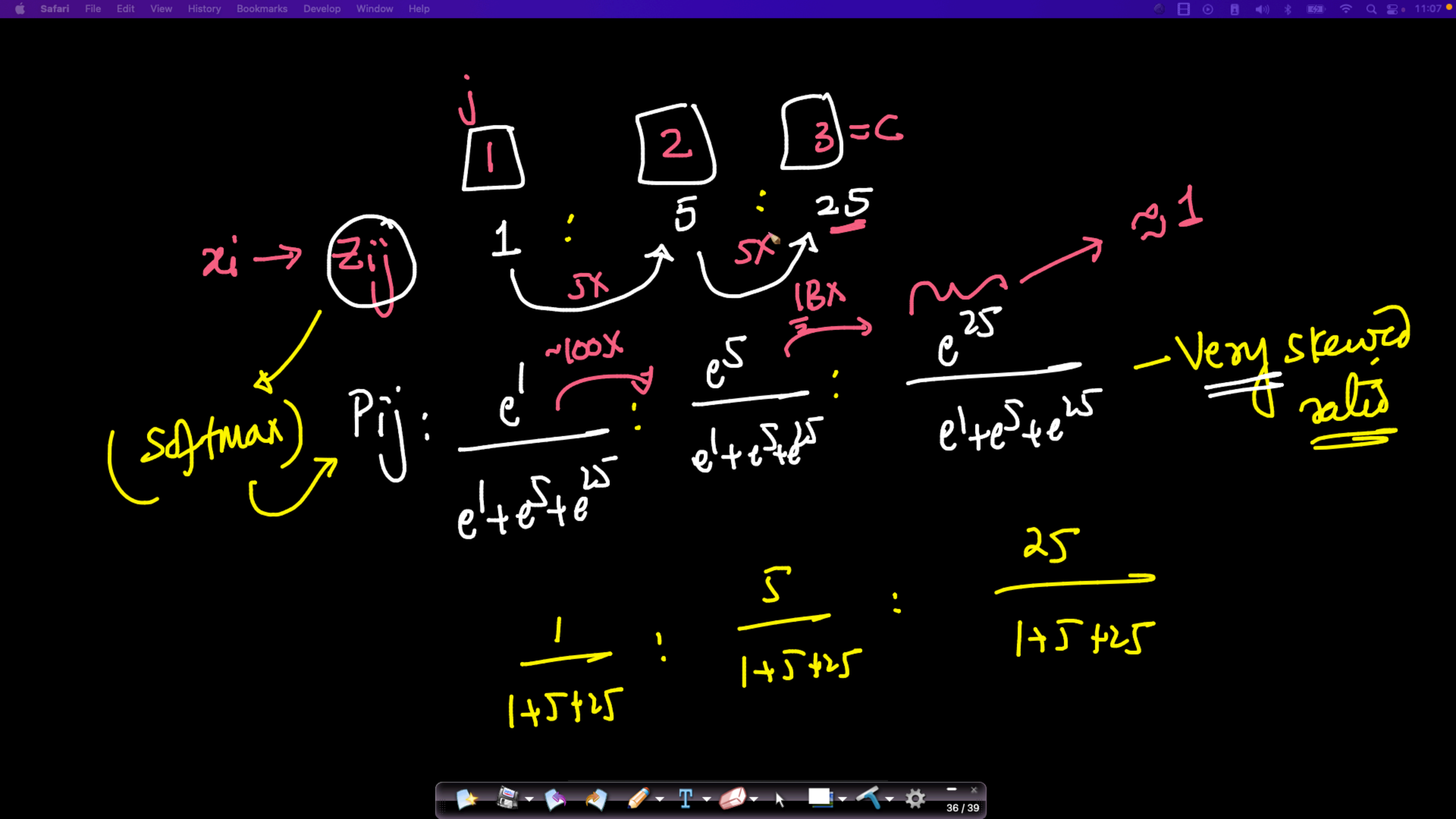
if $T=1$ eqn (1) = Softmax

where T is a temperature that is normally set to 1. Using a higher value for T produces a softer probability distribution over classes.

In the simplest for[illegible]odel by training it on

$$P_{ij}^{t} = \frac{\exp(z_{ij}^{t}/T)}{\sum_{j=1}^{C} \exp(z_{ij}^{t}/T)}$$

if $T=1$ → Softmax


j
1
2
3=C
1 : 5 : 25
5X
5X
$x_i \rightarrow z_{ij}$
(softmax)
P_{ij} :
$\frac{e^1}{e^1+e^5+e^{25}}$: $\frac{e^5}{e^1+e^5+e^{25}}$: $\frac{e^{25}}{e^1+e^5+e^{25}}$
~100X
148X
≈ 1
- Very skewed ratio
$\frac{1}{1+5+25}$: $\frac{5}{1+5+25}$: $\frac{25}{1+5+25}$

$$p_{ij} = \frac{\exp(\tilde{z}_{ij}/T)}{\sum_{j=1}^{C} \exp(\tilde{z}_{ij}/T)}$$

$T > 1$

$T = 5$

$T = 1 \ldots 20$

T: hyper-param

```
[27]  1 import math
      2
      3 z_i = [1,5,25] # outputs of the last layer before converting them into probabilities
      4 T=1 # Chnage Temperature to see the impact
      5 den=0 # of the probability equation
      6 for i in range(len(z_i)):
      7   den += math.exp(z_i[i]/T)
      8
      9 print(den)
```

```
72004899488.51732
```

```
1 p_i = [0]*len(z_i)
2 for i in range(len(z_i)):
3   p_i[i] = math.exp(z_i[i]/T)/den
4
5 print(p_i)
```

```
[3.775134536355449e-11, 2.061153618112392e-09, 0.9999999979010951]
```

Safari File Edit View History Bookmarks Develop Window Help

colab.research.google.com/drive/19TyCOV3PbGOPLumgphgwH_iWO1s3tx3V#scrollTo=bU0Q18vqoC_I

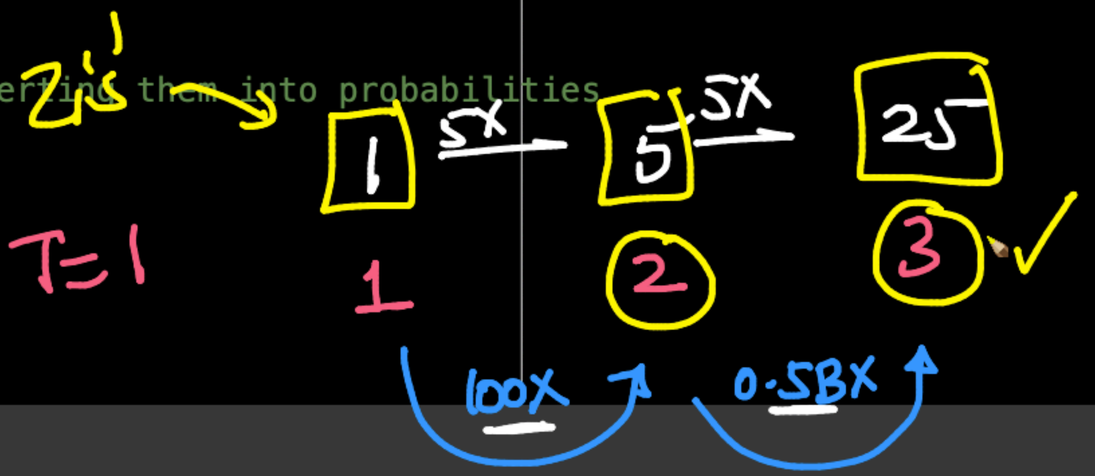
https://arxiv.org/pdf/2006.05525.pdf | Cross entropy - Wikipedia | https://arxiv.org/pdf/1503.02531.pdf | KnowledgeDistillationIntuition.ipynb - Colaboratory | https://arxiv.org/pdf/1910.01108.pdf

+ Code + Text

RAM Disk

[29]
```
1 import math
2
3 z_i = [1,5,25] # outputs of the last layer before converting them into probabilities
4 T=5 # Chnage Temperature to see the impact
5 den=0 # of the probability equation
6 for i in range(len(z_i)):
7   den += math.exp(z_i[i]/T)
8
9 print(den)
```

```
152.35284368919582
```



[30]
```
1 p_i = [0]*len(z_i)
2 for i in range(len(z_i)):
3   p_i[i] = math.exp(z_i[i]/T)/den
4
5 print(p_i)
```

```
[0.008016934430524097, 0.017842015696171828, 0.9741410498733041]
```

8×10^{-3} 17×10^{-3}

39 / 39

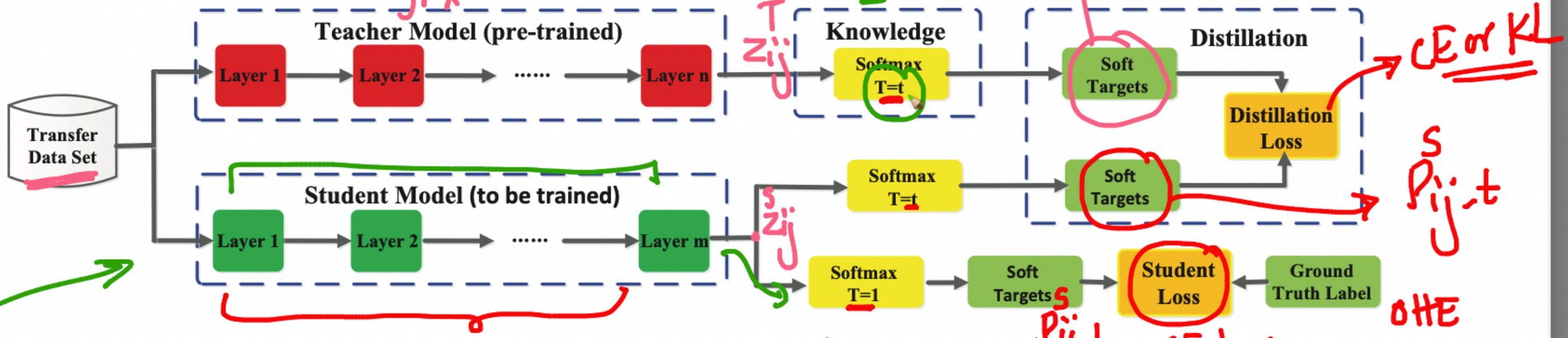


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c)	Parameters distribution	Multi-layer group	$\mathcal{L}_{CE}(.)$
AB (Heo et al., 2019c)	Activation boundaries	Pre-ReLU	$\mathcal{L}_2(.)$
Shen et al. (2019a)			$\mathcal{L}_2(.)$

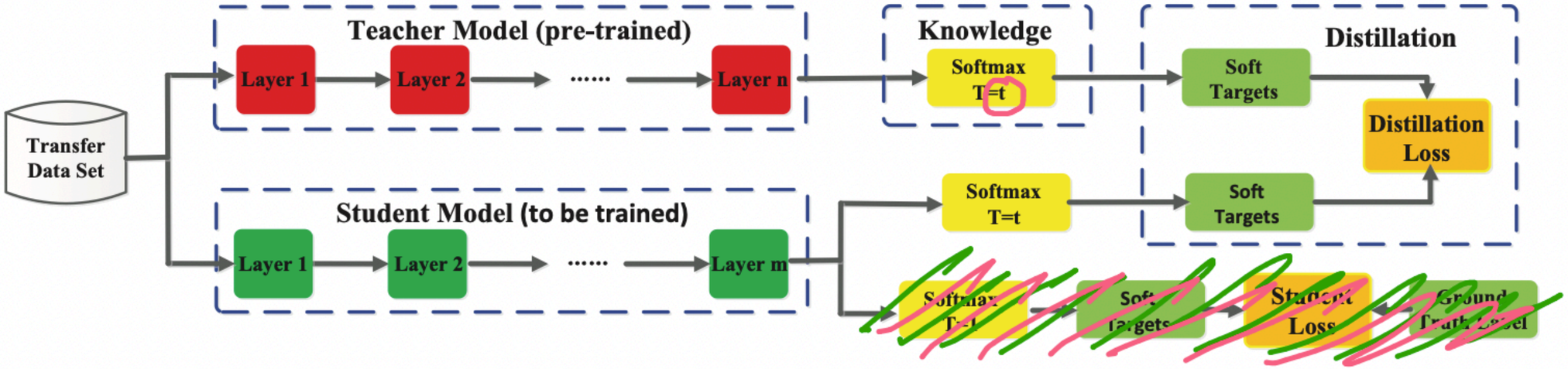


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c)	Parameters distribution	Multi-layer group	$\mathcal{L}_{CE}(.)$
AB (Heo et al., 2019c)	Activation boundaries	Pre-ReLU	$\mathcal{L}_2(.)$
Shen et al. (2019a)			$\mathcal{L}_2(.)$

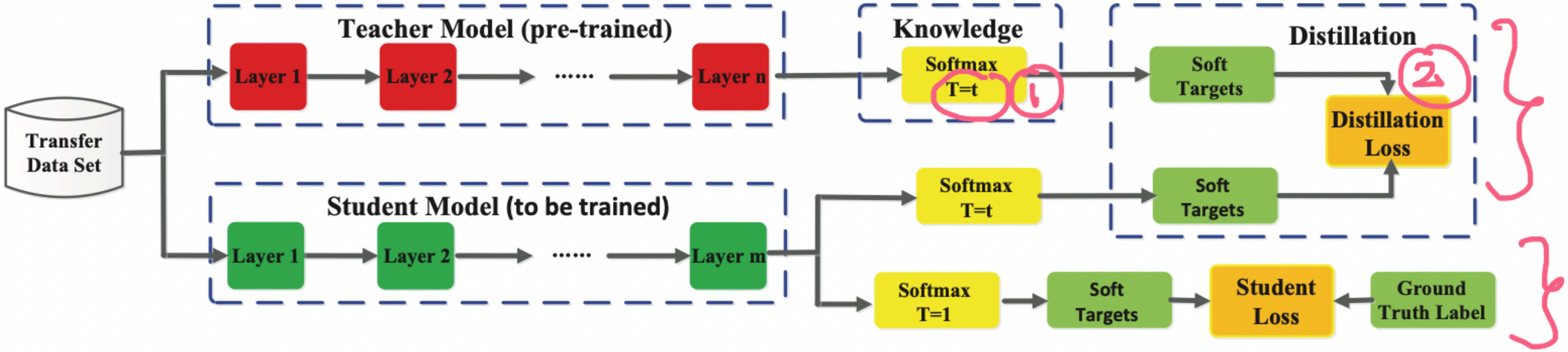


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c)	Parameters distribution	Multi-layer group	$\mathcal{L}_{CE}(.)$
AB (Heo et al., 2019c)	Activation boundaries	Pre-ReLU	$\mathcal{L}_2(.)$
Shen et al. (2019a)		er	$\mathcal{L}_2(.)$

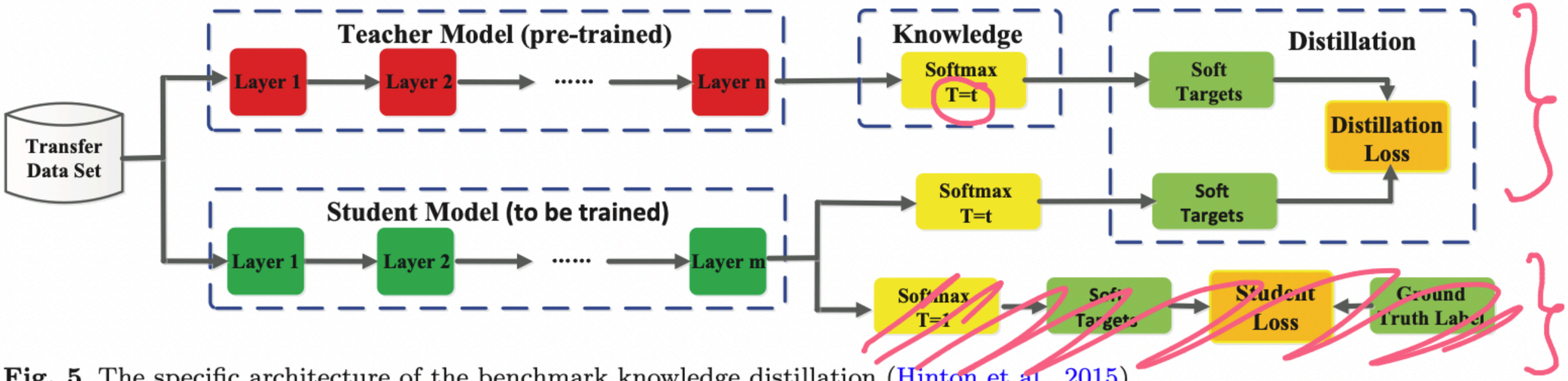


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c)	Parameters distribution	Multi-layer group	$\mathcal{L}_{CE}(.)$
AB (Heo et al., 2019c)	Activation boundaries	Pre-ReLU	$\mathcal{L}_2(.)$
Shen et al. (2019a)			$\mathcal{L}_2(.)$

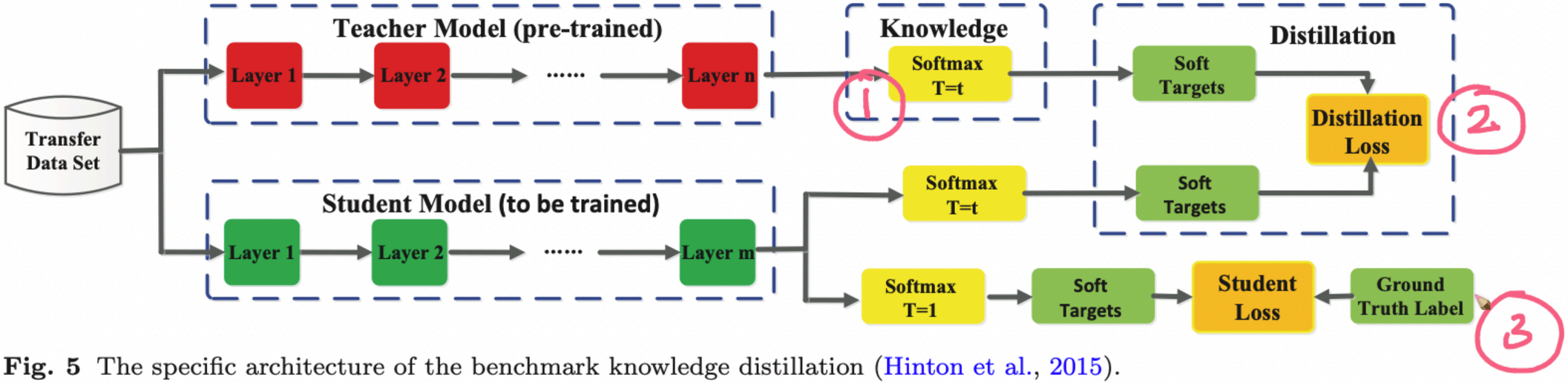


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

t=5

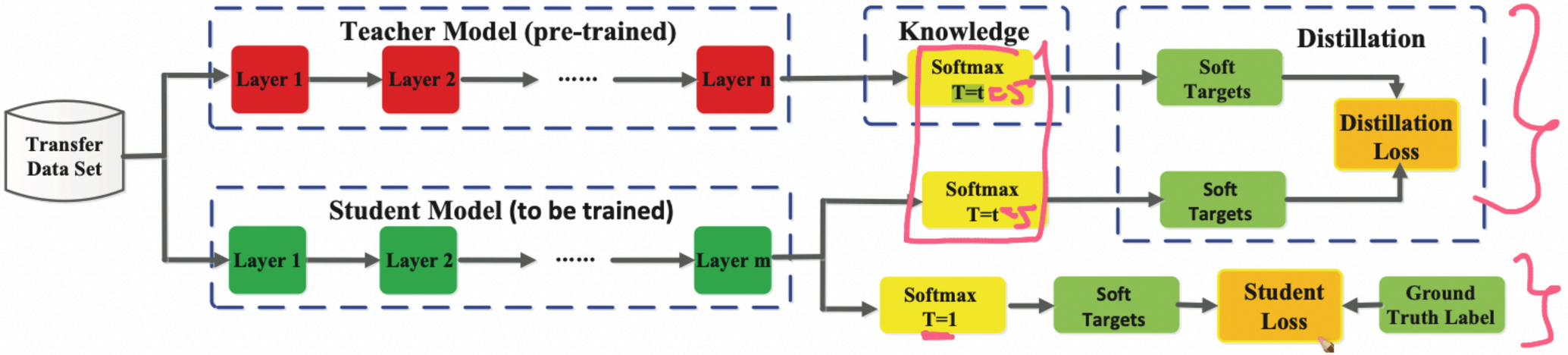


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

hyper-param
t=5

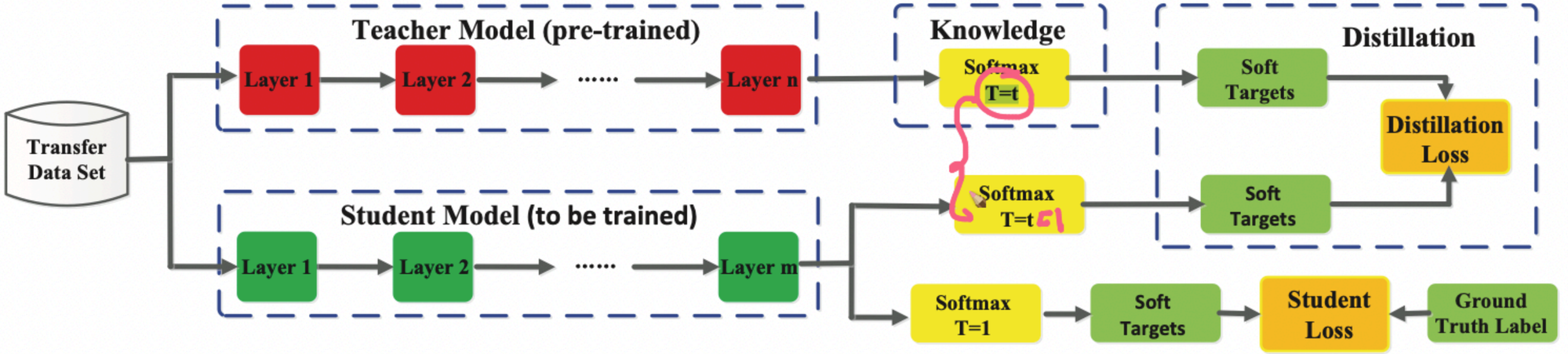


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

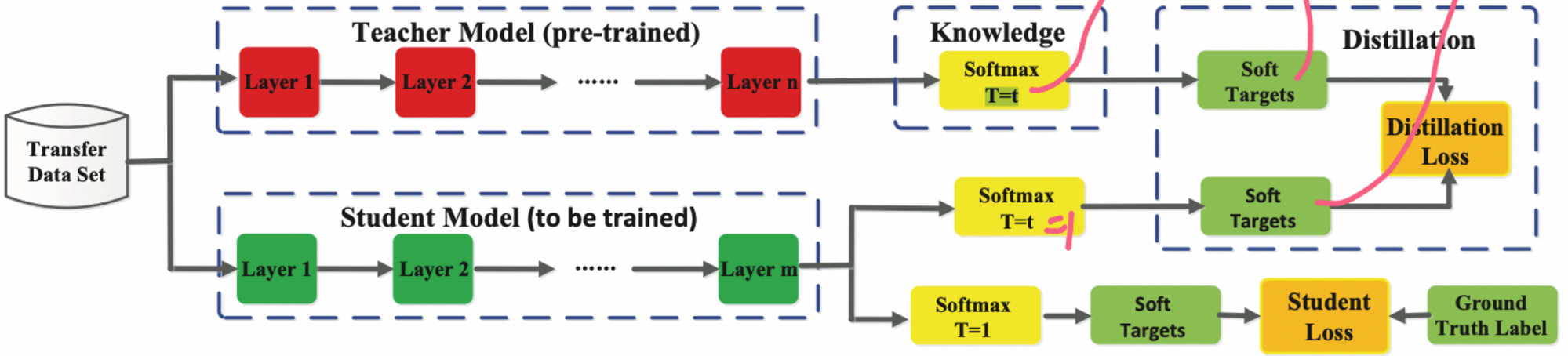


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

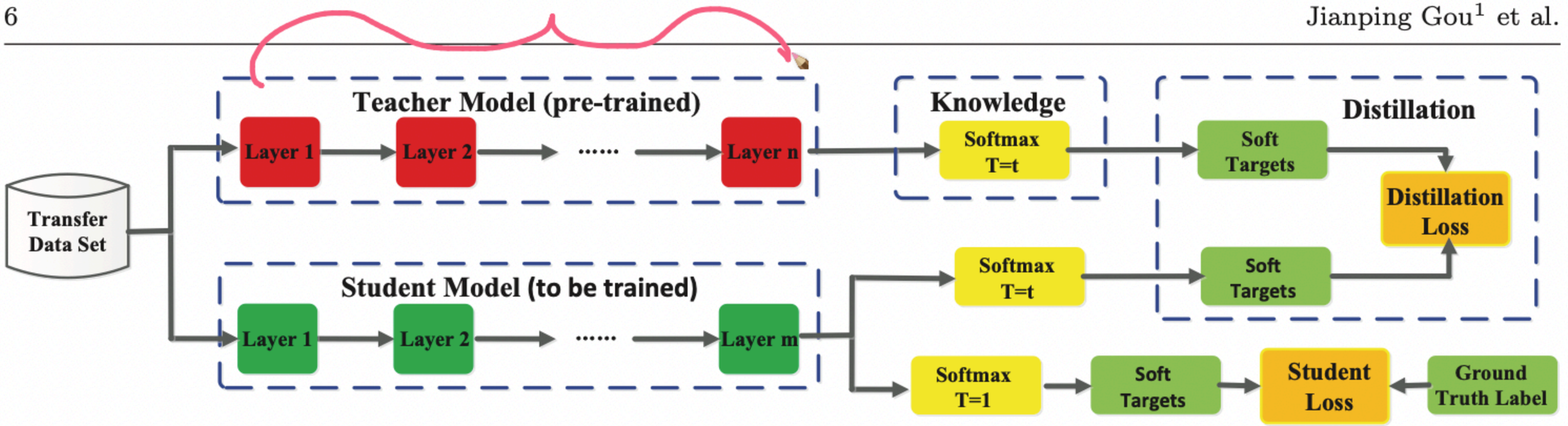


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

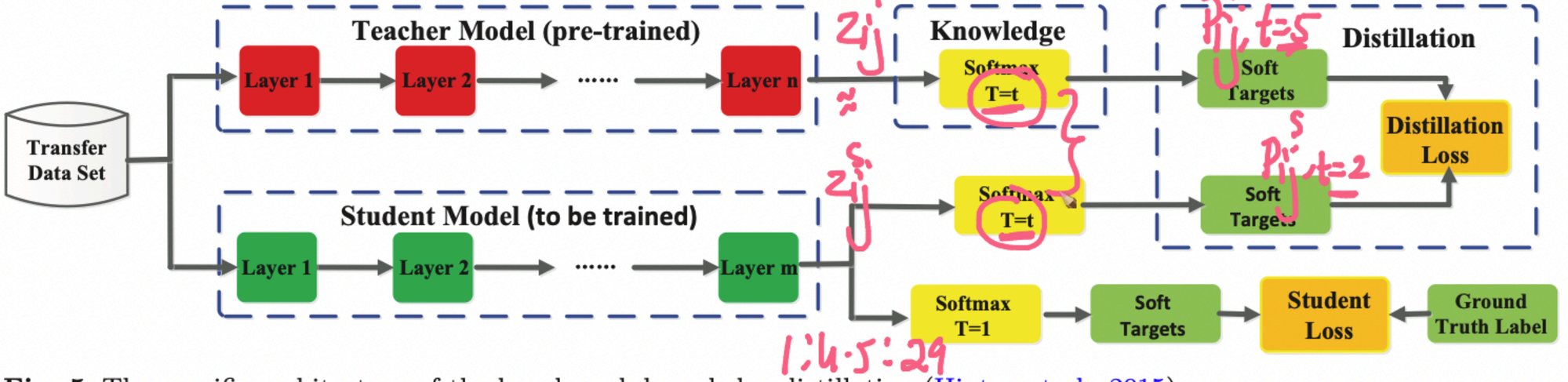


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

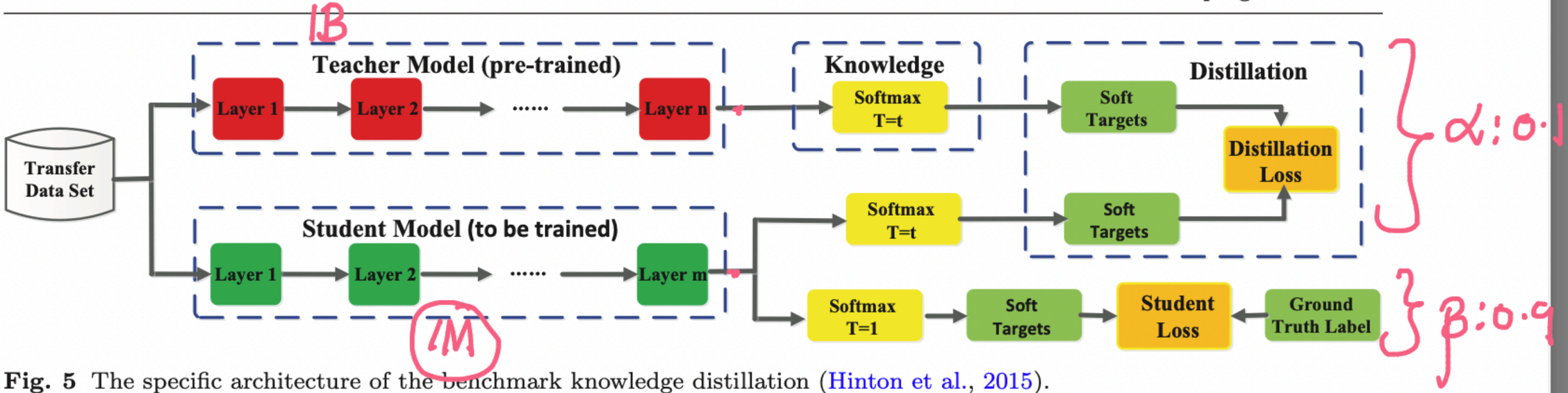


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

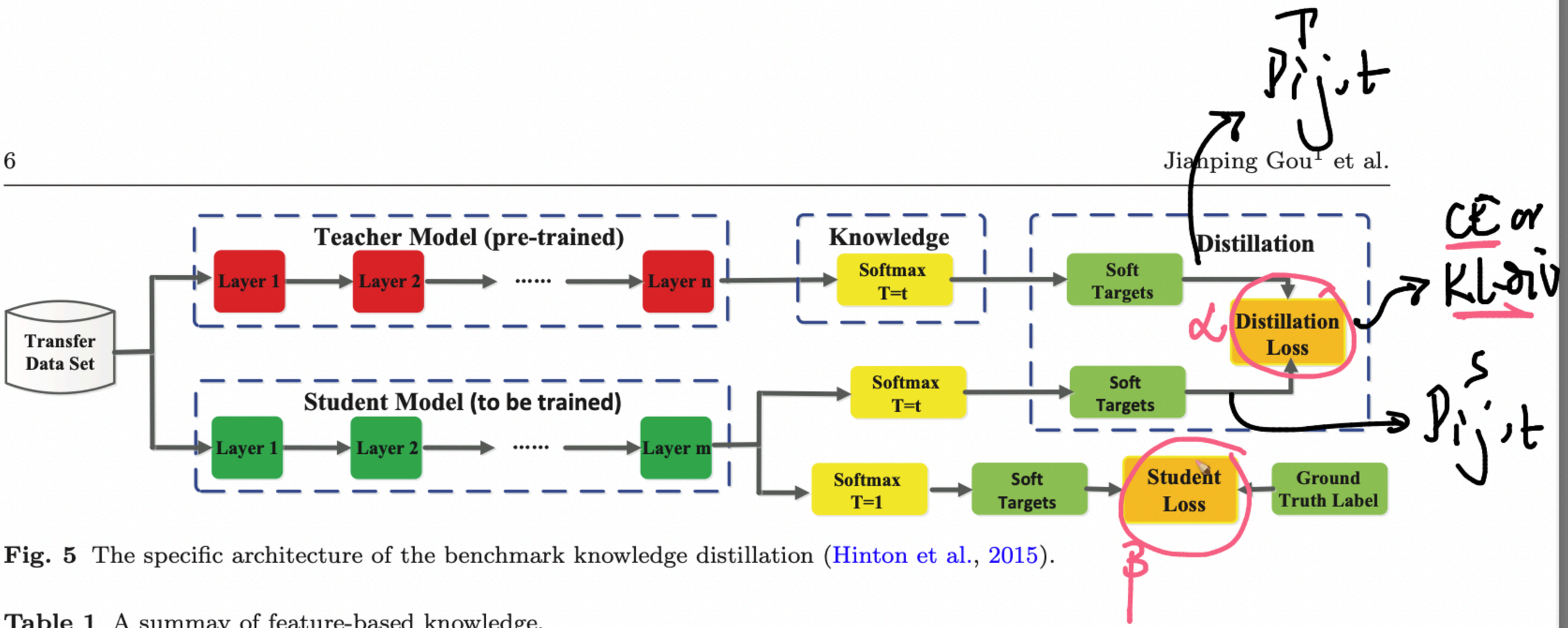


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

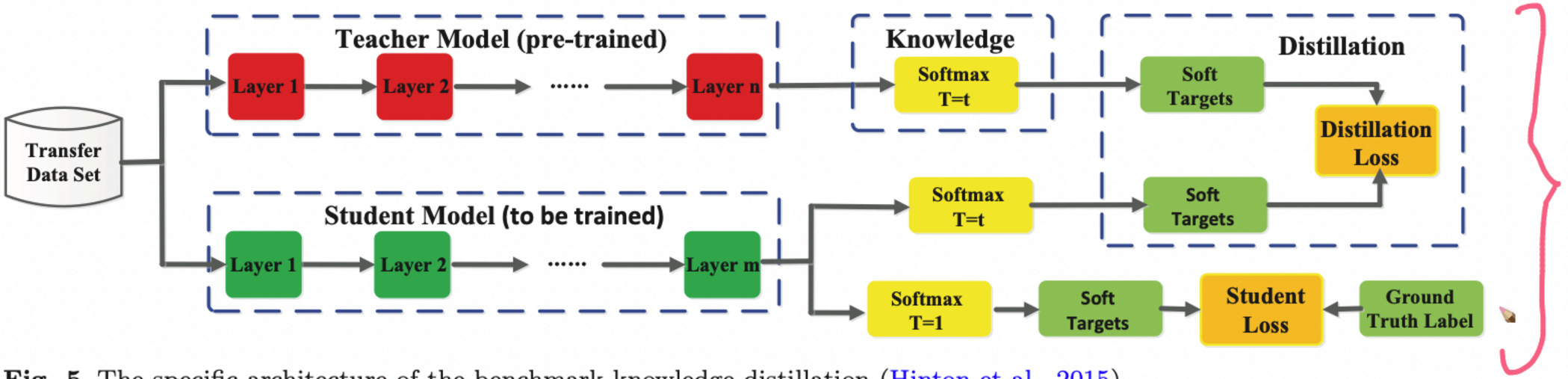


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

6 Conclusion and future work

We introduced DistilBERT, a general-purpose pre-trained version of BERT, 40% smaller, 60% faster, that retains 97% of the language understanding capabilities. We showed that a general-purpose language model can be successfully trained with distillation and analyzed the various components with an ablation study. We further demonstrated that DistilBERT is a compelling option for edge applications.

GLUE

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar S. Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke S. Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.

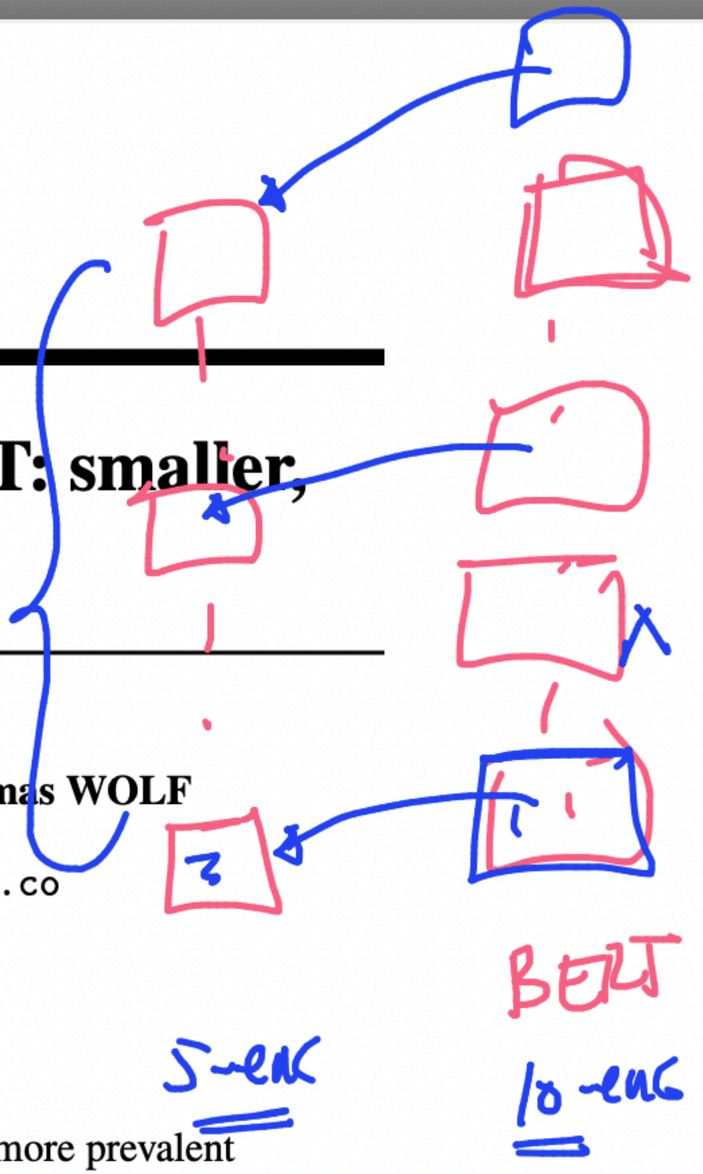
Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. Green ai. *ArXiv*, abs/1907.10597, 2019.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in nlp. In *ACL*, 2019.

Ashish Vaswani, No... Gomez, Lukasz Kaiser, and Illia Polosukh...

DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter

Victor SANH, Lysandre DEBUT, Julien CHAUMOND, Thomas WOLF
Hugging Face
{victor,lysandre,julien,thomas}@huggingface.co



Abstract

As Transfer Learning from large-scale pre-trained models becomes more prevalent in Natural Language Processing (NLP), operating these large models in on-the-edge and/or under constrained computational training or inference budgets remains challengi aller general-

Table 1: **DistilBERT retains 97% of BERT performance.** Comparison on the dev sets of the GLUE benchmark. ELMo results as reported by the authors. BERT and DistilBERT results are the medians of 5 runs with different seeds.

Model	**Score**	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST-2	STS-B	WNLI
ELMo	68.7	44.1	68.6	76.6	71.1	86.2	53.4	91.5	70.4	56.3
BERT-base	79.5	56.3	86.7	88.6	91.8	89.6	69.3	92.7	89.0	53.5
DistilBERT	77.0	51.3	82.2	87.5	89.2	88.5	59.9	91.3	86.9	56.3

Table 2: **DistilBERT yields to comparable performance on downstream tasks.** Comparison on downstream tasks: IMDb (test accuracy) and SQuAD 1.1 (EM/F1 on dev set). D: with a second step of distillation during fine-tuning.

Model	IMDb (acc.)	SQuAD (EM/F1)
BERT-base	93.46	81.2/88.5
DistilBERT	92.82	77.7/85.8
DistilBERT (D)	-	79.1/86.9

Table 3: **DistilBERT is significantly smaller while being constantly faster.** Inference time of a full pass of GLUE task STS-B (sentiment analysis) on CPU with a batch size of 1.

Model	# param. (Millions)	Inf. time (seconds)
ELMo	180	895
BERT-base	110	668
DistilBERT	66	410

Distillation We applied best practices for training BERT model recently proposed in Liu et al. [2019]. As such, DistilBERT is distilled on very large batches leveraging gradient accumulation (up to 4K examples per batch) using dynamic masking and without the next sentence prediction objective.

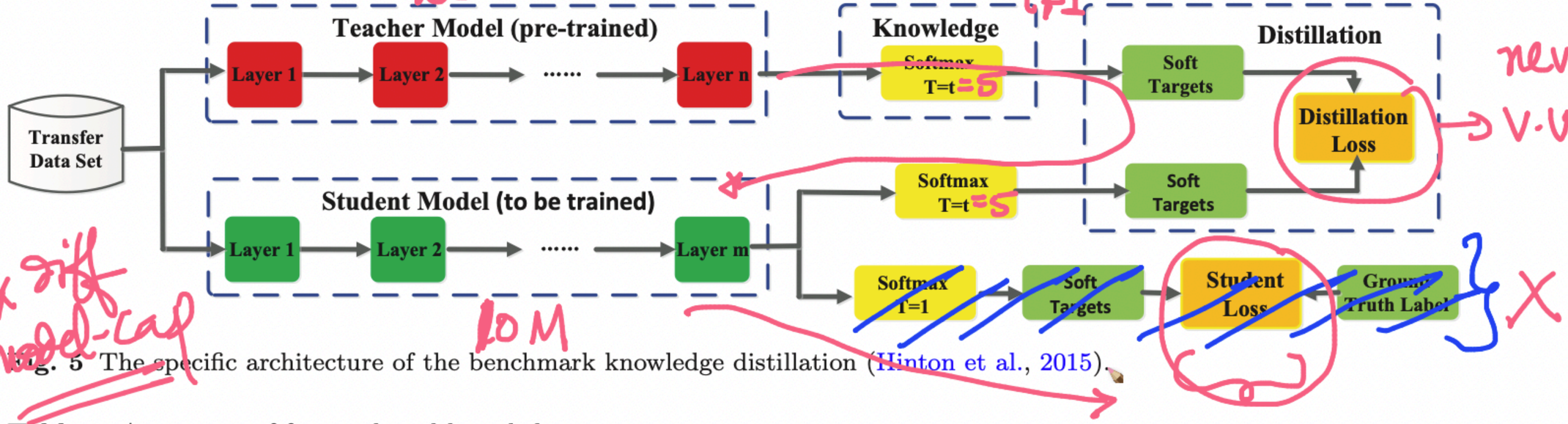


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

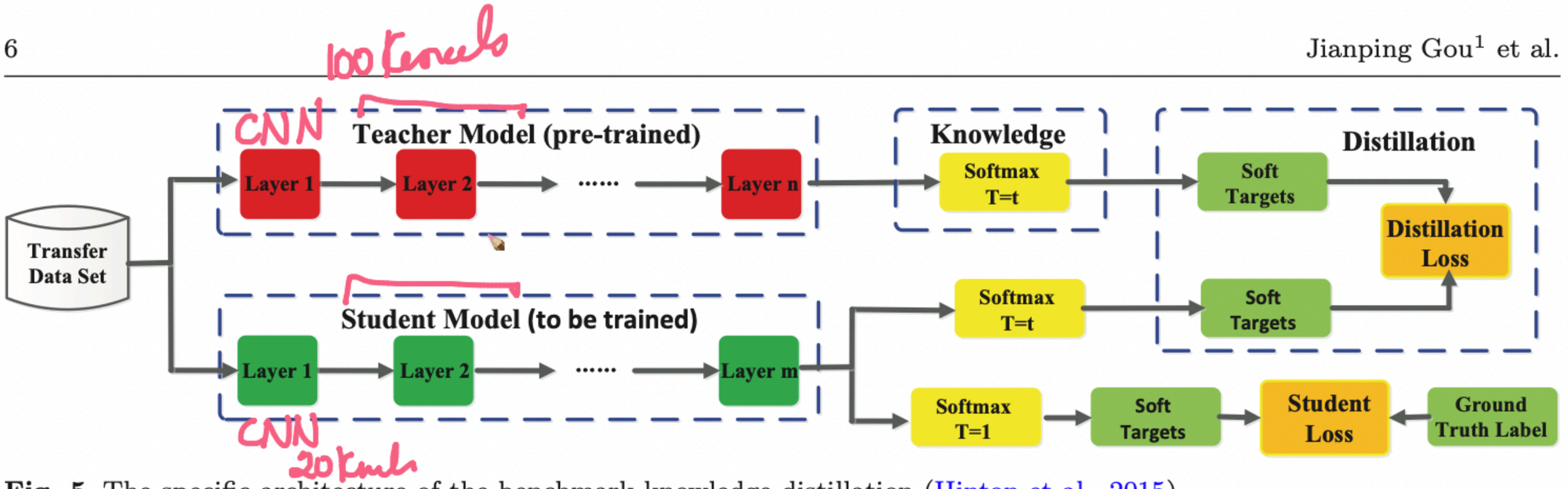


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

Distil Bert → BERT 12 → 5 encoder blocks enc-blocks

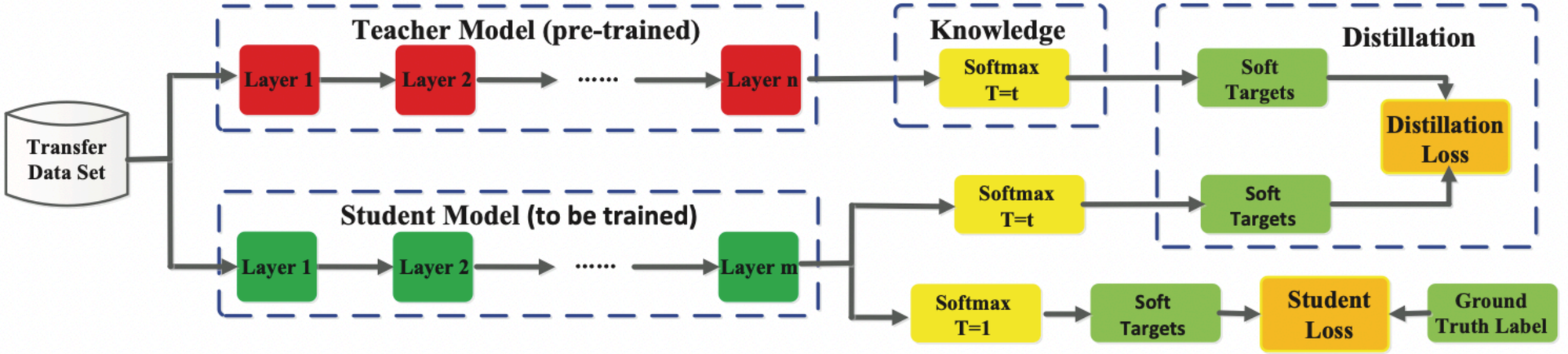


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

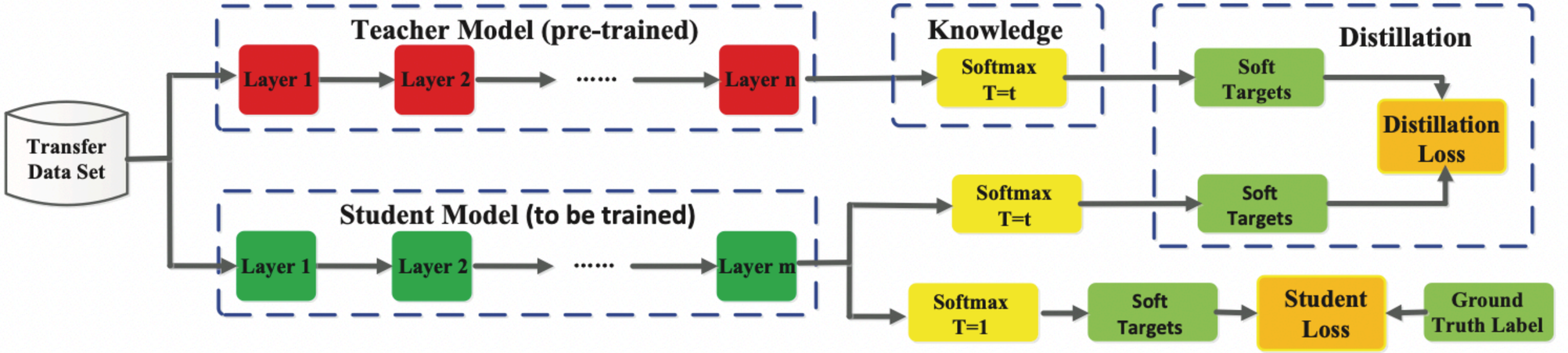


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

and teacher hidden states vectors.

3 DistilBERT: a distilled version of BERT

Student architecture In the present work, the student - DistilBERT - has the same general architecture as BERT. The *token-type embeddings* and the *pooler* are removed while the number of layers is reduced by a factor of 2. Most of the operations used in the Transformer architecture (*linear layer* and *layer normalisation*) are highly optimized in modern linear algebra frameworks and our investigations showed that variations on the last dimension of the tensor (hidden size dimension) have a smaller impact on computation efficiency (for a fixed parameters budget) than variations on other factors like the number of layers. Thus we focus on reducing the number of layers.

Student initialization In addition to the previously described optimization and architectural choices, an important element in our training procedure is to find the right initialization for the sub-network to converge. Taking advantage of the common dimensionality between teacher and student networks, we initialize the student from the teacher by taking one layer out of two.

---

[2] https://github.com/huggingface/transformers

[3] E.g. BERT-base's predictions for a masked token in "`I think this is the beginning of a beautiful [MASK]`" comprise two high probability tokens (*day* and *life*) and a long tail of valid predictions (*future*, *story*, *world*...).

and teacher hidden states vectors.

3 DistilBERT: a distilled version of BERT

Student architecture In the present work, the student - DistilBERT - has the same general architecture as BERT. The *token-type embeddings* and the *pooler* are removed while the number of layers is reduced by a factor of 2. Most of the operations used in the Transformer architecture (*linear layer* and *layer normalisation*) are highly optimized in modern linear algebra frameworks and our investigations showed that variations on the last dimension of the tensor (hidden size dimension) have a smaller impact on computation efficiency (for a fixed parameters budget) than variations on other factors like the number of layers. Thus we focus on reducing the number of layers.

Student initialization In addition to the previously described optimization and architectural choices, an important element in our training procedure is to find the right initialization for the sub-network to converge. Taking advantage of the common dimensionality between teacher and student networks, we initialize the student from the teacher by taking one layer out of two.

[2] https://github.com/huggingface/transformers

[3] E.g. BERT-base's predictions for a masked token in "`I think this is the beginning of a beautiful [MASK]`" comprise two high probability tokens (*day* and *life*) and a long tail of valid predictions (*future*, *story*, *world*...).

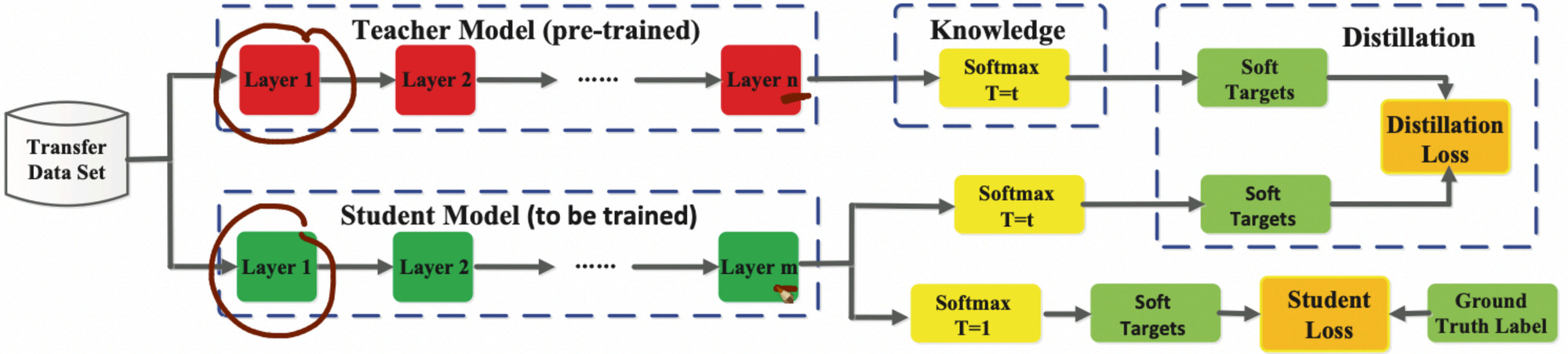
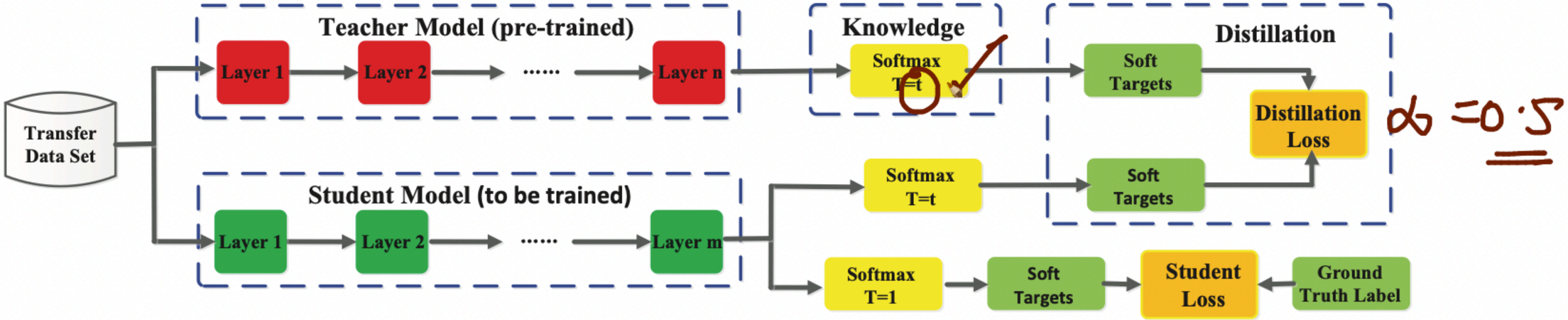


Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c)		group	$\mathcal{L}_{CE}(.)$

hyper-param ← $0 \leq \alpha \leq 1$



$\alpha = 0.5$

$(1-\alpha) = \beta$

Fig. 5 The specific architecture of the benchmark knowledge distillation (Hinton et al., 2015).

Table 1 A summay of feature-based knowledge.

Feature-based knowledge			
Methods	Knowledge Types	Knowledge Sources	Distillation losses
Fitnet (Romero et al., 2015)	Feature representation	Hint layer	$\mathcal{L}_2(.)$
NST (Huang and Wang, 2017)	Neuron selectivity patterns	Hint layer	$\mathcal{L}_{MMD}(.)$
AT (Zagoruyko and Komodakis, 2017)	Attention maps	Multi-layer group	$\mathcal{L}_2(.)$
FT (Kim et al., 2018)	Paraphraser	Multi-layer group	$\mathcal{L}_1(.)$
Rocket Launching (Zhou et al., 2018)	Sharing parameters	Hint layer	$\mathcal{L}_2(.)$
KR (Liu et al., 2019c		group	$\mathcal{L}_{CE}(.)$

Chrome File Edit View History Bookmarks Profiles Tab Window Help 12:03

Community Dashboard | Airmeet: Knowledge D | OpenCV: Camera Cali | Multiple View Geomet | 3D Object Detection | Buy Computer Vision: | Microsoft PowerPoint | Buy Computer Vision: | Book.html

airmeet.com/airmeets/9e96cdf0-25f2-11ed-a1a1-bb2c7a9977b7/summary

Airmeet

Knowledge Distillation in Deep Learning Completed

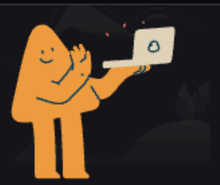
28 Aug 2022, 10:00 AM - 12:00 PM IST · Social Webinar

Share Event | Enter Event

Summary | Schedule | Registrations | Branding | Promotions | Analytics | Settings

Summary

View Landing Page



Post-event access is now available

Showcase your session recordings after the event has ended.

Enable Post-Event Access

Schedule (1 activity) View

28 Aug
10:00 am - 12:00 PM

Knowledge Distillation in Deep Learning
120 mins

Speakers (0) View

You haven't invited any speakers

Hosts (1) View

Event details Edit

28 Aug 2022, 10:00 AM - 12:00 PM IST

No description added

Anyone can enter via their unique link

Applied Roots

No contact email address added

Event settings

Status of all your event settings

- Session recording
- Leaderboard
- Event replay
- Live stream
- Stream into Airmeet
- Integrations

Social Webinar Registrations

54 Registrations
46 of 100 registrations left
Not allowing extra registrations beyond limit Edit

54%

64 / 64

Let us help you setup the event

Post event activities