Trustworthy Distillation? — High-Temperature Distillation as Double Regularization Plus Gradient Masking

Ruomu (Felix) Zou
Yale University
New Haven, CT 06511
felix.zou@yale.edu

Abstract

Knowledge distillation that transfers knowledge from a teacher model to a student model [7] has been proposed as a mechanism for defending deep neural networks against adversarial attacks when conducted at high temperatures [10]; however, it has also been argued that such distillation is merely a form of gradient masking and so is ultimately ineffective against stronger attacks [4, 5]. In this work, we explore the fundamental impacts both distillation and high-temperature training has on models. We find empirically that conducting distillation between teacher and student models of the same architecture at high temperatures has a double regularizing effect on the student model: one from the high temperature and a second from the soft labels in the distillation training process. Of course, if the student has an architecture smaller than the teacher, a third regularizing effect is present due to the inherent lower-complexity of the smaller student. Furthermore, we also confirm the numerical conclusions made in [4, 5] that high-temperature distillation conducts implicit gradient masking, and find that this is done by making the probabilities around decision boundary "harder" (akin to a step function rather than a sigmoid), thus rendering weak gradient-based attacks ineffective but still leaving the model vulnerable to stronger attacks. Finally, we confirm the observations made in [3, 1, 12] that distillation across different architectures produces students that are unfaithful to their teachers in terms of interpretability. However, we find that using teacher and student models with the same architecture improves this faithfulness significantly. Ultimately, despite the regularizing effects of distillation, its application to making models more trustworthy remains dubious at best. Our code is available at: https://github.com/zouruomu/TrustworthyDistillation

1 Problem Setting and Definition:

Knowledge distillation was originally introduced as a technique that can allow smaller and more computationally efficient models to learn the same information encoded in larger models or even ensembles of models [7, 2]. It has also been shown to have a regularizing effect largely due to the smoother labels used to train the student models [7, 14]. Distillation operates by training a teacher model (or models) normally, and then training the student model not on the original one-hot labels, but rather the soft probabilities generated by obtaining the teacher's predicted probabilities on the training set using a softmax with an additional temperature (T) parameter defined as follows:

$$p_i = \frac{e^{z_i/T}}{\sum_j e^{z_j/T}}$$

Preprint. Under review.

[10] proposes distilling a teacher network trained at some high temperature $T > 1$ (such as $T = 20$ or even $T = 100$) into a student network *of the same size and architecture* as a form of defense, and showed that it is effect against the Jacobian-Based Saliency Map Attack (JSMA) and the Fast Gradient Sign Method (FGSM) attack. However, more recent works have challenged the application of distillation to Trustworthy AI by arguing that it neither improves robustness [4, 5] nor produces interpretationally-faithful student models [3, 1, 12]. We aim to assess these claims and further investigate the nature of distillation.

We formulate our problem as three questions:

1. **What does distillation and, more generally, training at a high temperature actually do to the model?**
2. **Does high-temperature distillation actually make models more robust?**
3. **Does distilling a teacher into a student preserve the interpretations of the teacher?**

A brief summary of our findings for each questions are:

1. High-temperature distillation has a two-fold regularizing effect on the student model: one from the high temperature, and the other from the soft labels of the distillation training process. Furthermore, training any model at high-temperatures forces it to make the probabilities around its decision boundary harder.
2. Any training (distillation or not) at high-temperatures provides a first layer of defense due to hard decision boundary higher temperatures generate. This has a gradient masking effect that is sufficient to resist weak gradient-based attacks. Modifying the attack as was done in [4, 5] largely negates this advantage. The inherent regularization effect distillation has provides a second, albeit significantly weaker, defensive effect that stronger attacks cannot negate.
3. Distillation performed at any combination of teacher/student temperatures is generally unable to make the student possess the same interpretations as the teacher. However, distilling a teacher model into a student model with the same architecture and size makes the student interpretations *marginally* more faithful compared to using a student with a smaller architecture.

2 Relevance to Trustworthy AI:

We investigate both the *adversarial robustness* and *explainability* of models trained with various flavors of distillation and aim to assess the application of distillation on Trustworthy AI. Furthermore, the starting-point of our investigations is the distillation framework proposed in [10], which is covered in lecture. Finally, we use a variety of attacks such as FGSM, PGD, and Deepfool, as well as a variety of interpretability methods such as saliency maps, Grad-CAM, and Integrated Gradients, all of which are highly relevant to Trustworthy AI.

3 Description of related works:

The main work that inspired our explorations was [10] (and a short follow-up [9]), which proposed distillation (in the form of what the authors termed "Defensive Distillation") as a defense against adversarial attack. It proposes to train a teacher model with some high softmax temperature T (note that this differs from traditional knowledge distillation as formulated in [7], which uses a temperature of 1 for the teacher model) and distill its knowledge into a student model with the same architecture and size using soft labels.

Two other works that highly influenced our thinking were [4] and [5]. The original defensive distillation work [10] hypothesizes that distillation has an inherent "softening" effect that makes models trained with it less sensitive to perturbation and thus more adversarially robust. [4] and [5] empirically demonstrated that this hypothesis is false by presenting strong versions of regular attacks (which for the remainder of this report will be termed the "CW Strong" versions of regular attack) that can succeed on defensively distilled networks by only dividing their pre-softmax logits by T

without modifying anything else, thus showing that the defensive effects of distillation lies primarily in the high-temperature softmax and not some inherent aspect of the networks themselves.

Finally, in terms of interpretability, the works [3], [1], [12] all argue that distilled student models have interpretations that differ significantly from their teachers. These works drove our explorations in the explainability realm.

4 Proposed Approaches:

4.1 Distillation models and method:

In investigating the three questioned listed in the problem definition, we consider 20 (4 teacher, 16 student) different model training methods that represent different flavors of distillation and apply them to the three tasks below (in total we train 120 models: 20 for $CIFAR10$, 20 for $TSRD$ discussed below, and 80 models at 4 different temperatures for a synthetic toy dataset). These models address a broad spectrum of variations across model size (***Full*** or ***Reduced***) and training temperature ($T = 1$ or $T \geq 20$) for both teachers and students.

Specifically, we define the 4 following teachers models for each dataset:

1. FT_temp1 (Full-architecture Teacher trained with softmax temperature 1): This is how models are normally trained and conveniently serves as a baseline for other models.

2. RT_temp1 (Reduced-architecture Teacher trained with softmax temperature 1): This model has a smaller architecture (how much smaller is dependent on the dataset) than FT_temp1 but is otherwise trained the normal way.

3. FT_tempT (Full-architecture Teacher trained with softmax temperature T): This is an unconventional way of training a teacher model, as it forces the model's logits to be T-times greater than FT_temp1. [10] trains teacher models in this manner.

4. RT_tempT (Reduced-architecture Teacher trained with softmax temperature T): Smaller version of FT_tempT.

For each teacher, we define 4 student models in the same way (S stands for student): $FS\_temp1\_from\_\{teacher\}$, $RS\_temp1\_from\_\{teacher\}$, $FS\_tempT\_from\_\{teacher\}$, $RS\_tempT\_from\_\{teacher\}$. For example, $RS\_tempT\_from\_FT\_temp1$ would represent a Reduced-architecture Student trained with softmax temperature T on soft labels generated from a Full-architecture Teacher trained with softmax temperature 1. Note that, regardless of teacher/student training temperatures, all the soft labels are always generated by running inference on the teacher model with temperature T.

These 20 models encompass every conventional (and unconventional) way of doing knowledge distillation. For example, $RS\_tempT\_from\_FT\_temp1$ represents the most textbook-standard knowledge distillation described in [7], whereas $F\_tempT\_from\_FT\_tempT$ represents the formulation used in the defensive distillation paper [10].

4.2 Robustness/explainability evaluation methods and metrics:

To evaluate robustness, we run 6 attacks on each model. We first use the vanilla (weak) versions of the *FGSM*, *PGD*, and *Deepfool* attacks. Then we construct the stronger versions of these attacks by dividing the model logits by T as proposed in [4, 5] and attack the model again with each stronger attack. To evaluate performance, we use test set prediction accuracy from a sub-sample of the test set.

To evaluate explainability, we run 3 attribution methods (saliency map, Integrated Gradients, and Grad-CAM) on all models. To evaluate the faithfulness of a given student model to its teacher model, we compute the normalized (to $[0, 1]$) Hadamard product sum (effectively a kind of cosine-similarity) between the student attributions and the teacher attributions on images from a sub-sample of the test set. The more faithful the attributions are, the higher the value between 0 and 1.

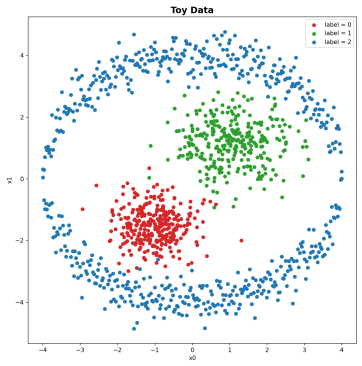

Figure 1: The full synthetic toy dataset.

4.3 Note on novelty:

Aside from traditional textbook distillation and defensive distillation, our 20 flavors encompass a wide range of unconventional distillation methods. This makes our approach highly novel in its range of exploration. Furthermore, in the next section, we also run experiments on a novel dataset.

5 Experiments:

We first run (with $T = \{10, 40, 70, 100\}$) the many flavors of distillation on a synthetic toy dataset to get an idea for what it does to a model. Then, we fix the temperature at the value recommended by [10] ($T = 20$) and run the many flavors again on two other datasets: *CIFAR10* and *TSRD* (Traffic Sign Recognition Dataset).

5.1 Synthetic toy dataset:

To allow us to visualize model decision heatmaps and boundaries on a screen, we create a 2-dimensional toy dataset that consists of 3 classes labeled $0 - 2$ organized as two Gaussian clusters within a ring. The entire dataset can be visualized in Figure 1. We train $20 \times 4 = 80$ models on this dataset, corresponding to 4 copies of the 20 flavors above at the different temperatures $T = \{10, 40, 70, 100\}$. Below are the notable results and analyses (the complete set of results can be seen in the appendix). The "Full" architecture used for this data is a three-layer-MLP with 32 hidden neurons, and the "Reduced" architecture is another three-layer-MLP with only 16 hidden neurons.

5.1.1 High-temperature training hardens decision boundaries:

We will proceed to answer the question of why training models at high temperatures can be used as a form of defense against weaker attackers. Figure 2 plots the trained decision heatmaps/boundaries of the baseline teacher model (trained at normal temperature), the high-temperature teacher, and the defensively distilled student. We observe that the normal temperature baseline transitions from predicting one class to another *smoothly*, since there is a sizable "twilight zone" where the boundary is blurry (i.e. where there is significant probability of being in either class). On the other hand, for the models trained at high temperatures – whether distilled (student) or not (teacher) – the transition is much harder, almost like a discrete decision boundary: the predicted probabilities are very close to one-hot vectors. Indeed, if the soft boundary of the normal-temperature base is likened to a sigmoid function, the harder boundaries of the high-temperature models can be likened to an almost-step-function.

Why does this happen? When a model is trained at the high temperature T in its softmax function, to minimize the cross-entropy loss, it is in turn forced to make the logits T times larger/more extreme so that the values actually being exponentiated (recall the logits are divided by T before passed to exp) are comparable in scale to using low temperatures. These larger logits, when the softmax temperature

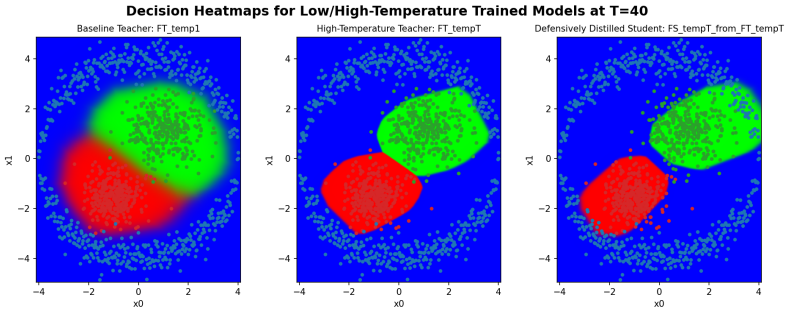

Figure 2: High-temperature training makes decision boundaries "harder". The RGB channels of each pixel is the model's predicted probability of being in each of the red/green/blue classes.

is set back to 1 at inference time and the logits are no longer divided by T, in turn make all the softmax outputs close to one-hot vectors since exponentiation favors higher inputs much more than lower inputs. This hardening of the decision boundary is equivalent to the gradient masking effect [4, 5] discussed. If all the probability vectors are near one-hot with zeros everywhere except the predicted class, adversaries have no meaningful gradients to exploit for non-predicted classes. Indeed, the other experiments will empirically confirm that most of the defensive capability of defensive distillation in [10] comes from the high-temperature training rather than the distillation itself: *one does not need distillation for defense against weak attacks*.

Lastly, note that, although both the high-temperature teacher and student have harder decision boundaries, the distilled student model appears to be more regularized (the regions for each class have more straight edges instead of smoother curves, signalling a smaller number of hidden neurons that contribute to it). We will now investigate this.

5.1.2 High-temperature distillation has a two-fold regularizing effect:

Now we will turn to the impact of distillation on the decision boundary shapes themselves (not just their hardness). Figure 3 shows what the decision heatmap/boundaries are for the same two teacher (FT_tempT) and student ($FS\_tempT\_from\_FT\_tempT$) models trained at temperatures varying from $T = 10$ to $T = 100$.

We first note that, since all these models are trained with high temperatures, the decision heatmaps/boundaries appear very hard for the reason discussed in the previous section. What

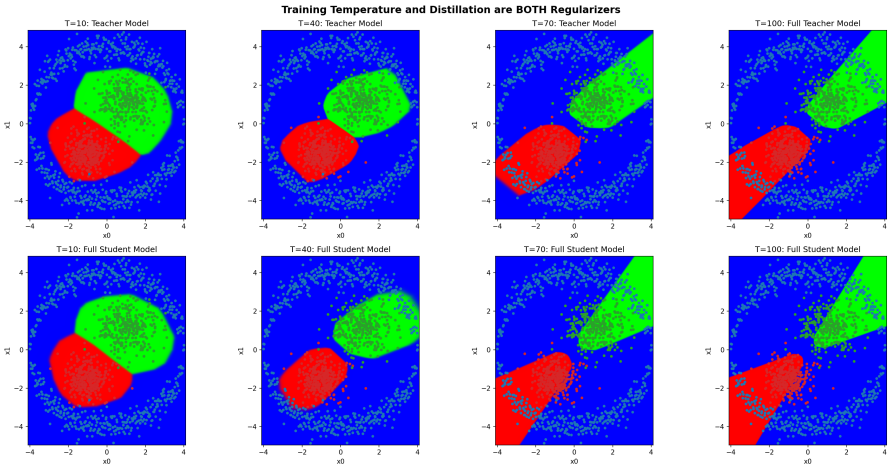

Figure 3: Higher training temperatures leads to more regularized decision boundaries (left to right). The presence of distillation also leads to more regularized decision boundaries (top to bottom).

is more interesting is the fact that, as one goes from top-left to bottom-right, the decision boundaries gets more and more regularized, featuring fewer and fewer piece-wise linear lines defining each region. Indeed, when temperatures get high, the distilled student model has decision regions that aren't even closed. For reference, training the baseline non-distilled teacher network at temperature 1 with dropout applied results in similar decision boundaries as the $T = 70$ and $T = 100$ student networks.

What is even more interesting is that the regularization effect is not solely accounted for by either temperature or presence of distillation alone. Indeed, we observe that, for the same temperature, student models trained with distillation exhibit more regularization than their teachers, and that, within the students/teachers, networks with higher temperature exhibit more regularization. This shows that temperature and the presence of distillation *both* have a regularizing effect.

Why is this the case? Regarding the presence of distillation, it has been widely studied that smoother labels have a regularizing effect in training [14, 8], and the presence of distillation implies that the student model is trained with the smooth probabilities from the teacher model, thereby resulting in regularization. However, this effect alone does not account for the observations made: the teacher model was always trained on hard labels, why does it also get more regularized as temperature increases? This is harder to answer, and we leave a thorough derivation for future work. Intuitively, we suspect that this is due to the fact that, compared to $T = 1$ training, the softmax outputs passed to cross-entropy are significantly more uniform, leading to the singular gradient signal (in the case of one-hot labels) from the label class being less informative/granular.

However, despite the regularizing effects of high-temperature distillation, past works have shown that regularization helps little in terms of adversarial robustness [13, 6]. We also verify this in our next experiments. Ultimately, the primary defensive capability in defensive distillation is still the harder decision boundary.

5.2 CIFAR10 dataset:

To compare our results with other works such as [10, 4, 5], we run all 20 flavors of distillation on the CIFAR10 dataset at the fixed temperature $T = 20$ (as recommended in [10]). Figure 4 displays the training loss curves for all 20 models.

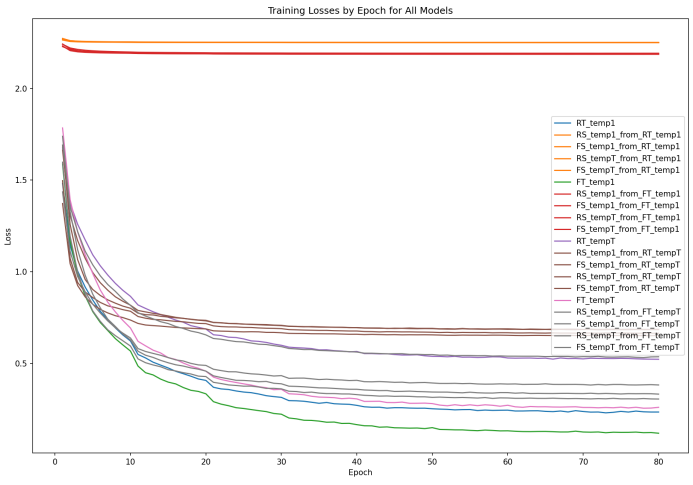

Figure 4: Training loss curves for all 20 models on CIFAR10.

For the "Full" architecture, we use the same CNN architecture used in [10], [9], [4], and [5]. With it, we achieve the same baseline accuracies as previous works (around 80% for testing and 98% for training, previous works all overfit this model and so we do as well). The "Reduced" architecture removes every other layer of this "Full" architecture and halves the sizes of the remaining layers.

5.2.1 Infeasibility of using normally-trained teachers at high distillation temperatures:

From figure 4, we observe that the 8 student models corresponding to teacher models that are trained normally (i.e. at temperature 1) – the red and orange curves – were not able to train much at all. This is because normally trained teachers have regular (non-enlarged) magnitude logits which, passed through a high-temperature softmax for distillation, results in a soft label that is essentially the uniform distribution. As such almost-uniform soft labels convey almost no information to the student, the student model is unable to learn. This finding explains why the authors of [10] needed to train the teacher model at high temperatures and cannot use the standard distillation procedure.

In general, when distilling under the high temperature necessary for the decision boundary to become hard enough to resist weak attacks, the teacher model must be trained with the same high temperature to ensure that the generated soft labels are informative.

5.2.2 Adversarial robustness:

Attacking All Models on the Testing Dataset Using Random Sample of 1000 Datapoints (Top 1 Accuracy)

RT_temp1:	RS_temp1_from_RT_temp1:	FS_temp1_from_RT_temp1:	RS_tempT_from_RT_temp1:	FS_tempT_from_RT_temp1:
* No Attack: 77.7%	* No Attack: 76.0%	* No Attack: 78.3%	* No Attack: 77.0%	* No Attack: 78.3%
* FGSM: 2.7%	* FGSM: 7.5%	* FGSM: 14.0%	* FGSM: 5.8%	* FGSM: 10.5%
* FGSM (CW): 2.7%	* FGSM (CW): 7.5%	* FGSM (CW): 14.0%	* FGSM (CW): 10.7%	* FGSM (CW): 12.5%
* PGD: 0.0%	* PGD: 0.5%	* PGD: 0.7%	* PGD: 0.6%	* PGD: 0.3%
* PGD (CW): 0.0%	* PGD (CW): 0.5%	* PGD (CW): 0.7%	* PGD (CW): 1.0%	* PGD (CW): 1.0%
* DeepFool: 3.1%	* DeepFool: 2.5%	* DeepFool: 2.0%	* DeepFool: 2.7%	* DeepFool: 1.9%
* DeepFool (CW): 3.1%	* DeepFool (CW): 2.5%	* DeepFool (CW): 2.0%	* DeepFool (CW): 2.3%	* DeepFool (CW): 1.5%

FT_temp1:	RS_temp1_from_FT_temp1:	FS_temp1_from_FT_temp1:	RS_tempT_from_FT_temp1:	FS_tempT_from_FT_temp1:
* No Attack: 83.4%	* No Attack: 79.0%	* No Attack: 83.6%	* No Attack: 77.6%	* No Attack: 82.2%
* FGSM: 9.7%	* FGSM: 5.1%	* FGSM: 10.0%	* FGSM: 9.4%	* FGSM: 8.3%
* FGSM (CW): 9.7%	* FGSM (CW): 5.1%	* FGSM (CW): 10.0%	* FGSM (CW): 7.4%	* FGSM (CW): 9.5%
* PGD: 0.0%	* PGD: 0.1%	* PGD: 0.0%	* PGD: 0.1%	* PGD: 0.1%
* PGD (CW): 0.0%	* PGD (CW): 0.1%	* PGD (CW): 0.0%	* PGD (CW): 0.4%	* PGD (CW): 0.1%
* DeepFool: 2.3%	* DeepFool: 3.8%	* DeepFool: 1.0%	* DeepFool: 3.6%	* DeepFool: 1.3%
* DeepFool (CW): 2.3%	* DeepFool (CW): 3.8%	* DeepFool (CW): 1.0%	* DeepFool (CW): 3.6%	* DeepFool (CW): 1.1%

RT_tempT:	RS_temp1_from_RT_tempT:	FS_temp1_from_RT_tempT:	RS_tempT_from_RT_tempT:	FS_tempT_from_RT_tempT:
* No Attack: 76.1%	* No Attack: 75.9%	* No Attack: 79.3%	* No Attack: 74.4%	* No Attack: 79.6%
* FGSM: 25.1%	* FGSM: 5.6%	* FGSM: 10.6%	* FGSM: 25.6%	* FGSM: 31.7%
* FGSM (CW): 3.8%	* FGSM (CW): 5.6%	* FGSM (CW): 10.6%	* FGSM (CW): 5.8%	* FGSM (CW): 10.3%
* PGD: 16.4%	* PGD: 0.2%	* PGD: 0.4%	* PGD: 12.1%	* PGD: 8.2%
* PGD (CW): 0.4%	* PGD (CW): 0.2%	* PGD (CW): 0.4%	* PGD (CW): 1.0%	* PGD (CW): 0.4%
* DeepFool: 3.2%	* DeepFool: 3.1%	* DeepFool: 1.7%	* DeepFool: 3.1%	* DeepFool: 1.5%
* DeepFool (CW): 3.9%	* DeepFool (CW): 3.1%	* DeepFool (CW): 1.7%	* DeepFool (CW): 3.8%	* DeepFool (CW): 1.8%

FT_tempT:	RS_temp1_from_FT_tempT:	FS_temp1_from_FT_tempT:	RS_tempT_from_FT_tempT:	FS_tempT_from_FT_tempT:
* No Attack: 84.1%	* No Attack: 78.1%	* No Attack: 84.3%	* No Attack: 77.2%	* No Attack: 84.8%
* FGSM: 52.3%	* FGSM: 4.0%	* FGSM: 7.8%	* FGSM: 29.1%	* FGSM: 52.6%
* FGSM (CW): 7.3%	* FGSM (CW): 4.0%	* FGSM (CW): 7.8%	* FGSM (CW): 5.0%	* FGSM (CW): 8.0%
* PGD: 45.9%	* PGD: 0.2%	* PGD: 0.1%	* PGD: 20.1%	* PGD: 40.9%
* PGD (CW): 0.0%	* PGD (CW): 0.2%	* PGD (CW): 0.1%	* PGD (CW): 0.7%	* PGD (CW): 0.1%
* DeepFool: 2.8%	* DeepFool: 2.6%	* DeepFool: 2.9%	* DeepFool: 3.9%	* DeepFool: 2.3%
* DeepFool (CW): 3.2%	* DeepFool (CW): 2.6%	* DeepFool (CW): 2.9%	* DeepFool (CW): 3.2%	* DeepFool (CW): 2.3%

Figure 5: CIFAR10 test-set accuracies when attacked with multiple attacks.

We now turn to observing the adversarial robustness of all 20 models. Figure 5 shows a grand overview of the top-1 test set accuracies (estimated with a 1000-image random sample) for every one of the 20 models on CIFAR10.

We first note that the weak versions of the attacks are highly potent with networks trained normally, but have reduced effectiveness for networks trained with high temperatures. This aligned with the motivation of defensive distillation, but also fundamentally undermines it: the model FT_tempT (bottom-left), which was not distilled at all and only trained at a high temperature, is still able to be effective at defending against weak attacks (in this case it actually does slightly better than its defensively distilled student $FS\_tempT\_from\_FT\_tempT$ due to the latter's slightly softer boundaries from training on soft labels – it doesn't need to increase its logits as much to match an already soft distribution). This aligns with our previous finding that distillation itself does not constitute the main defense, but rather the hard decision boundaries resulting from high-temperature training.

However, we also note that, controlling for the hardness of the decision boundary, the double regularization effect of high-temperature distillation does in fact improve robustness, albeit only marginally. To control for hardness, we apply the strong attacks (the "CW" versions) which divides

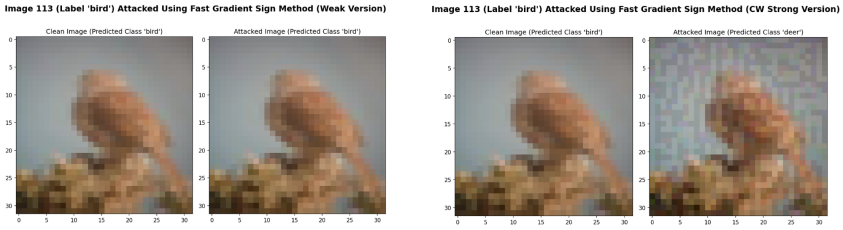

Figure 6: An illustration of the strong vs. weak versions of the FGSM attack on the same CIFAR10 image and the same defensively distilled network. The weak version did not have the gradients necessary to conduct the attack due to the hard decision boundaries.

all logits by T before passing them through the softmax as proposed by [4, 5]: doing so effectively softens up all the decision boundaries and scales down the hardness difference between FT_tempT and $FS\_tempT\_from\_FT\_tempT$ to a neglegible amount. Under the strong attacks, we observe that $FS\_tempT\_from\_FT\_tempT$ (bottom-right) performed marginally better than FT_tempT (bottom-left). Although the increase in accuracy is small, running the same test with multiple random seeds and the TSRD dataset below produced the same results, suggesting that the improved accuracy in the distilled student is not due to random chance but rather regularization.

5.2.3 Interpretability faithfulness:

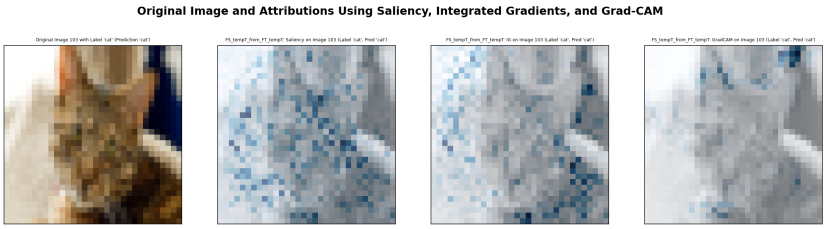

Figure 7: An example of attributions of the defensively distilled model using three interpretability methods on the same image.

Student model's adherence to teacher (in terms of attributed generated by different methods) on 1000 testing set images

RT_temp1:	RS_temp1_from_RT_temp1:	FS_temp1_from_RT_temp1:	RS_tempT_from_RT_temp1:	FS_tempT_from_RT_temp1:
	* Saliency: 0.644	* Saliency: 0.584	* Saliency: 0.661	* Saliency: 0.602
	* IG: 0.411	* IG: 0.371	* IG: 0.485	* IG: 0.412
	* GradCAM: 0.628	* GradCAM: 0.548	* GradCAM: 0.681	* GradCAM: 0.607
FT_temp1:	RS_temp1_from_FT_temp1:	FS_temp1_from_FT_temp1:	RS_tempT_from_FT_temp1:	FS_tempT_from_FT_temp1:
	* Saliency: 0.583	* Saliency: 0.569	* Saliency: 0.585	* Saliency: 0.572
	* IG: 0.266	* IG: 0.425	* IG: 0.274	* IG: 0.463
	* GradCAM: 0.432	* GradCAM: 0.673	* GradCAM: 0.462	* GradCAM: 0.679
RT_tempT:	RS_temp1_from_RT_tempT:	FS_temp1_from_RT_tempT:	RS_tempT_from_RT_tempT:	FS_tempT_from_RT_tempT:
	* Saliency: 0.669	* Saliency: 0.586	* Saliency: 0.71	* Saliency: 0.635
	* IG: 0.536	* IG: 0.394	* IG: 0.632	* IG: 0.479
	* GradCAM: 0.62	* GradCAM: 0.529	* GradCAM: 0.714	* GradCAM: 0.584
FT_tempT:	RS_temp1_from_FT_tempT:	FS_temp1_from_FT_tempT:	RS_tempT_from_FT_tempT:	FS_tempT_from_FT_tempT:
	* Saliency: 0.575	* Saliency: 0.631	* Saliency: 0.568	* Saliency: 0.661
	* IG: 0.279	* IG: 0.438	* IG: 0.221	* IG: 0.505
	* GradCAM: 0.401	* GradCAM: 0.575	* GradCAM: 0.341	* GradCAM: 0.652

Figure 8: The average interpretations adherence of each student to its corresponding teacher (leftmost column) by method.

Now we assess how faithful each student's interpretations are to those of its teacher. Figure 8 displays the average adherence (calculated over 1000 images as the normalized Hadamard product sum, effectively a cosine similarity) of each student to its teacher. Figure 9 displays the attributions for all students (rightmost 4 columns) and teachers (leftmost column) for a particular image. Note that the positioning for each model is the same across Figures 8 and 9.

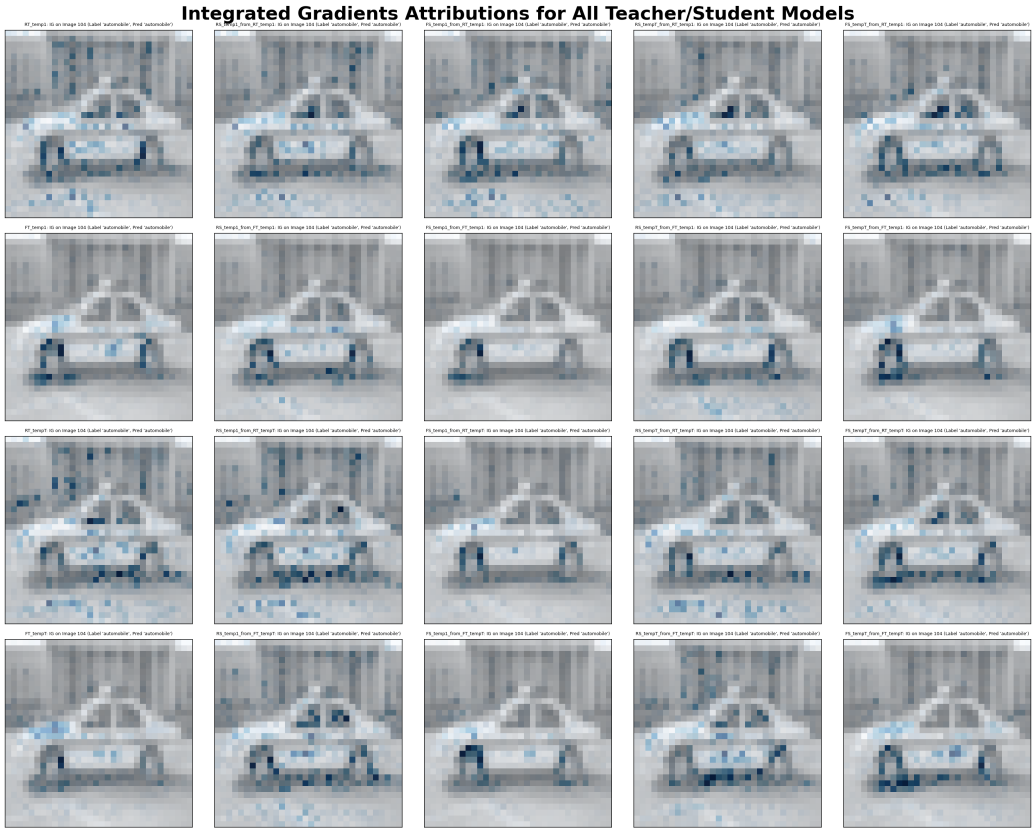

Figure 9: Integrated Gradients attributions of each teacher/student for a CIFAR10 image.

We note that, controlling for distillation temperatures, distilling across architecture yields interpretations that are highly varied, with smaller architectures generally exhibiting more noise. Distilling a model into the same architecture, however, yields much more similar interpretations.

5.3 TSRD dataset:

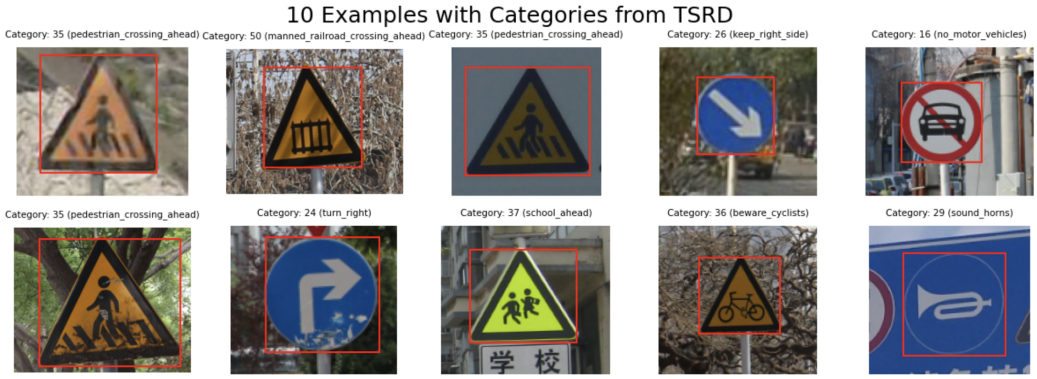

Figure 10: Examples from TSRD. The images are cropped to the red bounding box for training.

Finally, to examine how previous results generalize to non-trivial datasets, we run all 20 models on the real world Chinese Traffic Sign Recognition Dataset (TSRD). The goal of this dataset is to classify an image of a traffic sign into one of 58 total classes. Figure 10 displays 10 example images and their corresponding labels, and Figure 11 plots the occurrence counts of each class split between

the training and testing sets. Note that, as a pre-processing step, the images are cropped such that only the actual sign (inside the red bounding box in Figure 10) is seen by the network. Finally, Figure 12 displays the training loss curves for all models (much like the CIFAR10 case, teacher models trained normally fail to distill information into the students under high temperatures).

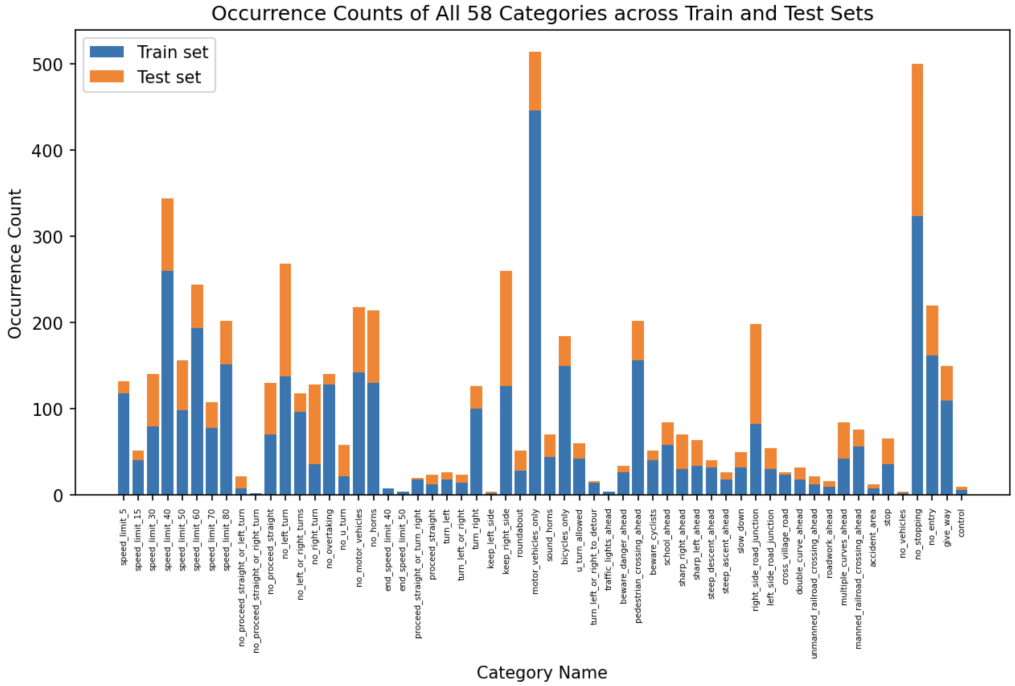

Figure 11: Class occurrence counts for all 58 classes in TSRD.

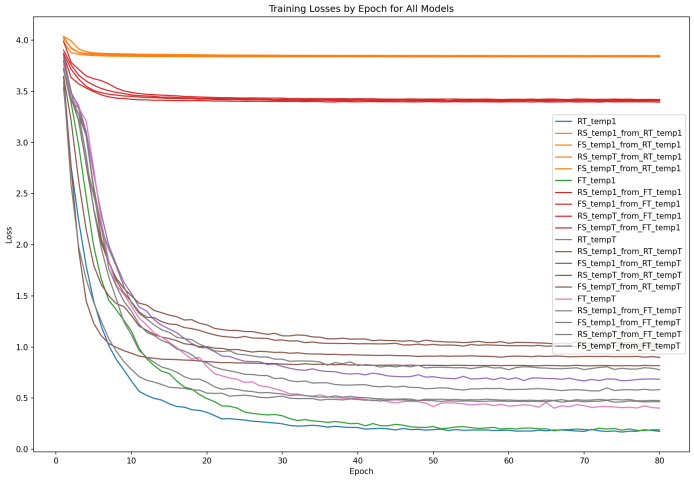

Figure 12: Training loss curves for all 20 models on TSRD.

The "Full" architecture is similar to the CIFAR10 full architecture, but is narrower and deeper with aggressive max-pooling (see released code for details). Again, the "Reduced" architecture removes every other layer of this "Full" architecture and halves the widths of the remaining layers.

5.3.1 Adversarial robustness:

We now turn to observing the adversarial robustness of all 20 models. Figure 13 shows a grand overview of the top-1 test set accuracies (estimated with a 300-image random sample) for every one of the 20 models on TSRD.

Attacking All Models on the Testing Dataset Using Random Sample of 300 Datapoints (Top 3 Accuracy)

RT_temp1:	RS_temp1_from_RT_temp1:	FS_temp1_from_RT_temp1:	RS_tempT_from_RT_temp1:	FS_tempT_from_RT_temp1:
* No Attack: 80.67%	* No Attack: 68.67%	* No Attack: 60.0%	* No Attack: 60.67%	* No Attack: 59.33%
* FGSM: 44.0%	* FGSM: 43.33%	* FGSM: 35.33%	* FGSM: 35.33%	* FGSM: 32.67%
* FGSM (CW): 44.0%	* FGSM (CW): 43.33%	* FGSM (CW): 35.33%	* FGSM (CW): 36.67%	* FGSM (CW): 32.67%
* PGD: 31.33%	* PGD: 23.67%	* PGD: 23.0%	* PGD: 24.67%	* PGD: 30.0%
* PGD (CW): 31.33%	* PGD (CW): 23.67%	* PGD (CW): 23.0%	* PGD (CW): 24.67%	* PGD (CW): 29.33%
FT_temp1:	RS_temp1_from_FT_temp1:	FS_temp1_from_FT_temp1:	RS_tempT_from_FT_temp1:	FS_tempT_from_FT_temp1:
* No Attack: 86.0%	* No Attack: 76.67%	* No Attack: 70.0%	* No Attack: 70.67%	* No Attack: 63.33%
* FGSM: 53.33%	* FGSM: 46.0%	* FGSM: 40.0%	* FGSM: 38.67%	* FGSM: 40.67%
* FGSM (CW): 53.33%	* FGSM (CW): 46.0%	* FGSM (CW): 40.0%	* FGSM (CW): 38.67%	* FGSM (CW): 39.67%
* PGD: 40.33%	* PGD: 19.0%	* PGD: 28.33%	* PGD: 31.67%	* PGD: 33.33%
* PGD (CW): 40.33%	* PGD (CW): 19.0%	* PGD (CW): 28.33%	* PGD (CW): 22.33%	* PGD (CW): 27.67%
RT_tempT:	RS_temp1_from_RT_tempT:	FS_temp1_from_RT_tempT:	RS_tempT_from_RT_tempT:	FS_tempT_from_RT_tempT:
* No Attack: 62.67%	* No Attack: 69.33%	* No Attack: 56.67%	* No Attack: 63.33%	* No Attack: 62.0%
* FGSM: 47.33%	* FGSM: 44.67%	* FGSM: 36.33%	* FGSM: 44.67%	* FGSM: 47.33%
* FGSM (CW): 34.33%	* FGSM (CW): 44.67%	* FGSM (CW): 36.33%	* FGSM (CW): 38.67%	* FGSM (CW): 36.33%
* PGD: 35.33%	* PGD: 32.67%	* PGD: 27.33%	* PGD: 36.0%	* PGD: 43.0%
* PGD (CW): 28.33%	* PGD (CW): 32.67%	* PGD (CW): 27.33%	* PGD (CW): 30.67%	* PGD (CW): 24.67%
FT_tempT:	RS_temp1_from_FT_tempT:	FS_temp1_from_FT_tempT:	RS_tempT_from_FT_tempT:	FS_tempT_from_FT_tempT:
* No Attack: 87.33%	* No Attack: 86.0%	* No Attack: 85.33%	* No Attack: 69.33%	* No Attack: 79.33%
* FGSM: 60.0%	* FGSM: 41.33%	* FGSM: 44.67%	* FGSM: 49.33%	* FGSM: 60.67%
* FGSM (CW): 38.0%	* FGSM (CW): 41.33%	* FGSM (CW): 44.67%	* FGSM (CW): 39.67%	* FGSM (CW): 40.67%
* PGD: 54.67%	* PGD: 27.0%	* PGD: 32.33%	* PGD: 40.0%	* PGD: 55.0%
* PGD (CW): 30.33%	* PGD (CW): 27.0%	* PGD (CW): 32.33%	* PGD (CW): 33.67%	* PGD (CW): 32.67%

Figure 13: TSRD test-set accuracies when attacked with multiple attacks.

Reassuringly, we observe the same trends on TSRD as on CIFAR10:

1. Weak attacks are ineffective at attacking high-temperature-trained network.
2. Controlling for training temperature, the defensively distilled student model is able to resist the strong attacks slightly better than the high-temperature teacher due to the double regularization effect of high-termperature distillation.

5.3.2 Interpretability faithfulness:

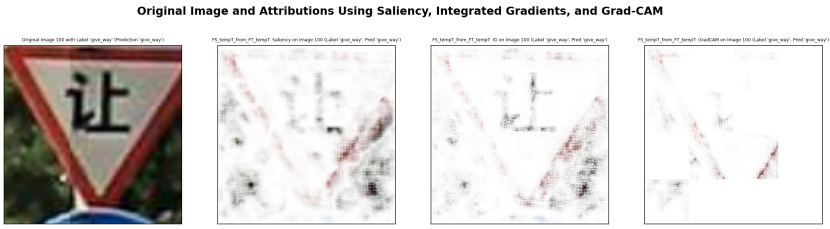

Figure 14: An example of attributions of the defensively distilled model using three interpretability methods on the same image.

Now we assess how faithful each student's interpretations are to those of its teacher. Just as in the CIFAR10 case, Figure 15 displays the average adherence (calculated over 100 images as the normalized Hadamard product sum, effectively a cosine similarity) of each student to its teacher. Figure 16 displays the Integrated Gradients attributions for all students (rightmost 4 columns) and teachers (leftmost column) for a particular image. Note that the positioning for each model is the same across Figures 15 and 16.

Once again, we can make the same observations as CIFAR10 that distillation across model architectures yields different interpretations whereas distillation within the same architecture preserves

Student model's adherence to teacher (in terms of attributed generated by different methods) on 100 testing set images

RT_temp1:	RS_temp1_from_RT_temp1: * Saliency: 0.527 * IG: 0.311 * GradCAM: 0.501	FS_temp1_from_RT_temp1: * Saliency: 0.384 * IG: 0.218 * GradCAM: 0.299	RS_tempT_from_RT_temp1: * Saliency: 0.563 * IG: 0.393 * GradCAM: 0.552	FS_tempT_from_RT_temp1: * Saliency: 0.378 * IG: 0.228 * GradCAM: 0.346
FT_temp1:	RS_temp1_from_FT_temp1: * Saliency: 0.356 * IG: 0.114 * GradCAM: 0.145	FS_temp1_from_FT_temp1: * Saliency: 0.434 * IG: 0.316 * GradCAM: 0.417	RS_tempT_from_FT_temp1: * Saliency: 0.319 * IG: 0.126 * GradCAM: 0.238	FS_tempT_from_FT_temp1: * Saliency: 0.406 * IG: 0.208 * GradCAM: 0.367
RT_tempT:	RS_temp1_from_RT_tempT: * Saliency: 0.599 * IG: 0.504 * GradCAM: 0.684	FS_temp1_from_RT_tempT: * Saliency: 0.334 * IG: 0.187 * GradCAM: 0.297	RS_tempT_from_RT_tempT: * Saliency: 0.612 * IG: 0.556 * GradCAM: 0.782	FS_tempT_from_RT_tempT: * Saliency: 0.306 * IG: 0.152 * GradCAM: 0.251
FT_tempT:	RS_temp1_from_FT_tempT: * Saliency: 0.28 * IG: 0.052 * GradCAM: 0.24	FS_temp1_from_FT_tempT: * Saliency: 0.538 * IG: 0.424 * GradCAM: 0.565	RS_tempT_from_FT_tempT: * Saliency: 0.259 * IG: 0.044 * GradCAM: 0.288	FS_tempT_from_FT_tempT: * Saliency: 0.56 * IG: 0.438 * GradCAM: 0.572

Figure 15: The average interpretations adherence of each student to its corresponding teacher (leftmost column) by method.

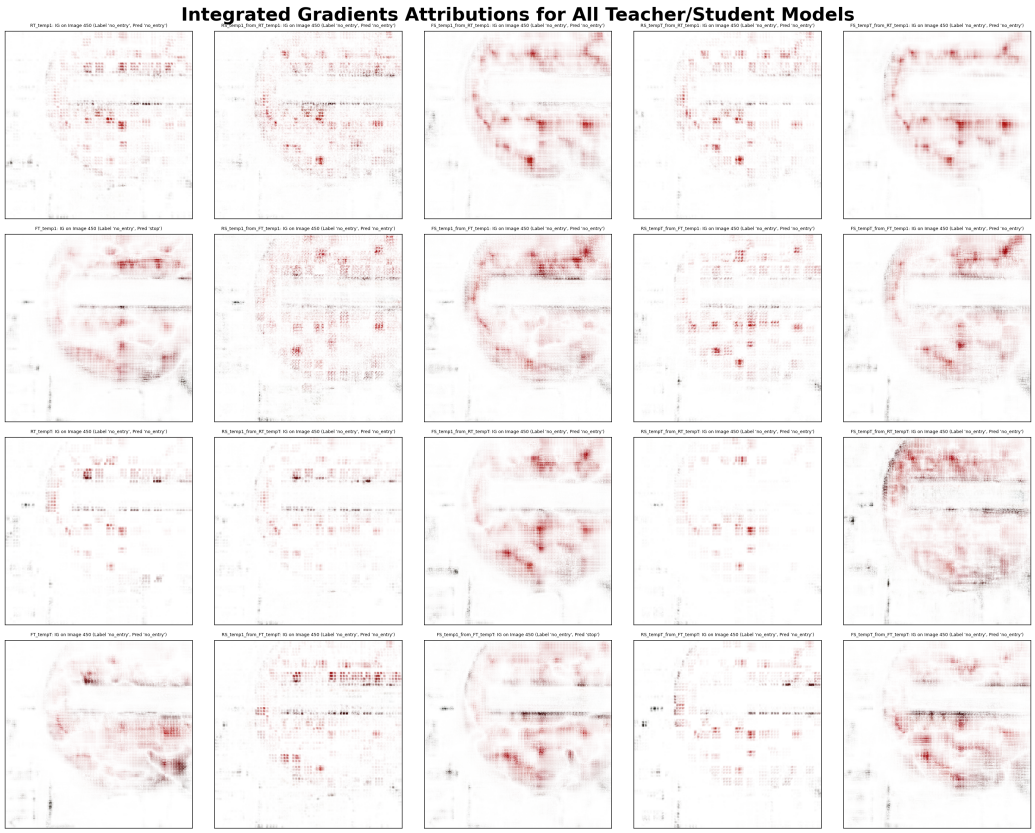

Figure 16: Integrated Gradients attributions of each teacher/student for a TSRD image.

faithfulness more. This also shows that our prior observations are capable of generalizing across datasets.

6 Conclusion:

In this project we thoroughly explore the nature of high-temperature distillation and assess its applications to Trustworthy AI. We find that it has a double-regularizing effect on the decision boundaries of the student networks and also makes those boundaries harder, with the hardening

being the primary mechanism by which it can be a defense against weak adversarial attack. We also find that the interpretations carry through distillation somewhat well for teacher/student models with the same architecture, and not at all when the architectures are different. Our findings have implications on the perception of high-temperature distillation as being not completely useless in the realm of Trustworthy AI despite the abundance of work successfully attacking it [4, 5, 11]. That said, ultimately, the role high-temperature distillation plays in improving the robustness/interpretability of model is still marginal at best.

Finally, the smallness of the datasets used present a limitation on our approach. Further exploration might extend our approach to datasets such as ImageNet.

References

[1] R. Alharbi, M. N. Vu, and M. T. Thai. Learning interpretation with explainable knowledge distillation. In *2021 IEEE International Conference on Big Data (Big Data)*, pages 705–714, Los Alamitos, CA, USA, dec 2021. IEEE Computer Society. doi: 10.1109/BigData52589.2021.9671988. URL https://doi.ieeecomputersociety.org/10.1109/BigData52589.2021.9671988.

[2] J. Ba and R. Caruana. Do deep nets really need to be deep? In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/file/ea8fcd92d59581717e06eb187f10666d-Paper.pdf.

[3] D. Batic, G. Tanoni, L. Stankovic, V. Stankovic, and E. Principi. Improving knowledge distillation for non-intrusive load monitoring through explainability guided learning. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023. doi: 10.1109/ICASSP49357.2023.10095109.

[4] N. Carlini and D. A. Wagner. Towards evaluating the robustness of neural networks. *CoRR*, abs/1608.04644, 2016. URL http://arxiv.org/abs/1608.04644.

[5] N. Carlini and D. A. Wagner. Defensive distillation is not robust to adversarial examples. *CoRR*, abs/1607.04311, 2016. URL http://arxiv.org/abs/1607.04311.

[6] I. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv 1412.6572*, 12 2014.

[7] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network, 2015.

[8] R. Müller, S. Kornblith, and G. E. Hinton. When does label smoothing help? *CoRR*, abs/1906.02629, 2019. URL http://arxiv.org/abs/1906.02629.

[9] N. Papernot and P. D. McDaniel. On the effectiveness of defensive distillation. *CoRR*, abs/1607.05113, 2016. URL http://arxiv.org/abs/1607.05113.

[10] N. Papernot, P. D. McDaniel, X. Wu, S. Jha, and A. Swami. Distillation as a defense to adversarial perturbations against deep neural networks. *CoRR*, abs/1511.04508, 2015. URL http://arxiv.org/abs/1511.04508.

[11] M. Soll, T. Hinz, S. Magg, and S. Wermter. Evaluating defensive distillation for defending text processing neural networks against adversarial examples. In I. V. Tetko, V. Kůrková, P. Karpov, and F. Theis, editors, *Artificial Neural Networks and Machine Learning – ICANN 2019: Image Processing*, pages 685–696, Cham, 2019. Springer International Publishing. ISBN 978-3-030-30508-6.

[12] T. Sun, H. Chen, G. Hu, and C. Zhao. Explainability-based knowledge distillation. 2023. URL http://dx.doi.org/10.2139/ssrn.4460609.

[13] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. 12 2013.

[14] L. Yuan, F. E. H. Tay, G. Li, T. Wang, and J. Feng. Revisit knowledge distillation: a teacher-free framework. *CoRR*, abs/1909.11723, 2019. URL http://arxiv.org/abs/1909.11723.

7 Appendix:

7.1 Toy dataset results:

7.1.1 $T = 10$:

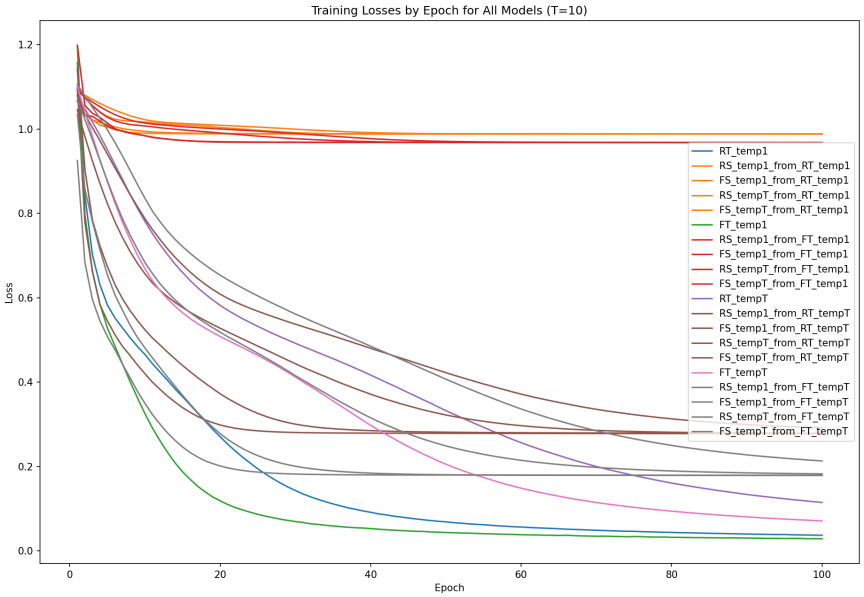

Figure 17: Training loss plots for all models ($T = 10$).

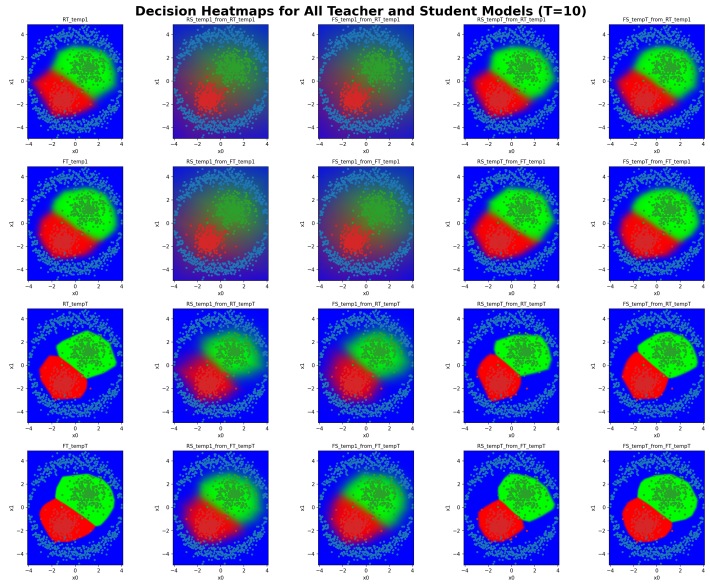

Figure 18: Decision heatmaps for all models ($T = 10$).

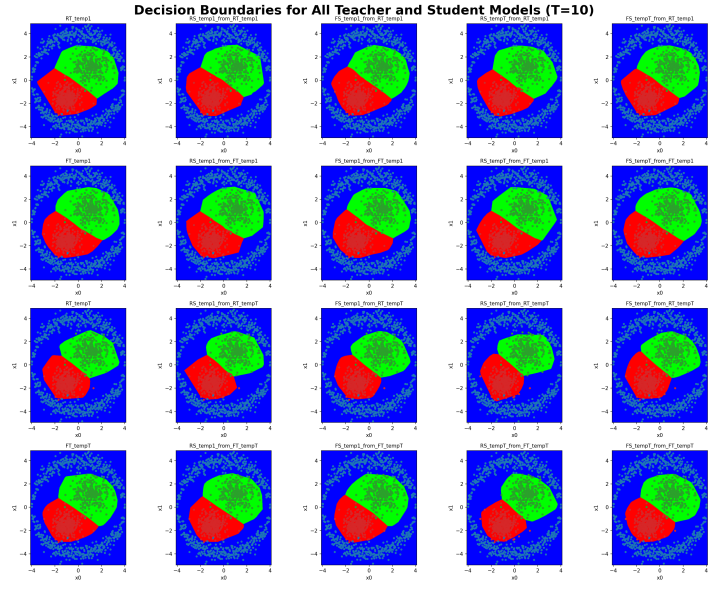

Figure 19: Rounded decision boundaries for all models ($T = 10$).

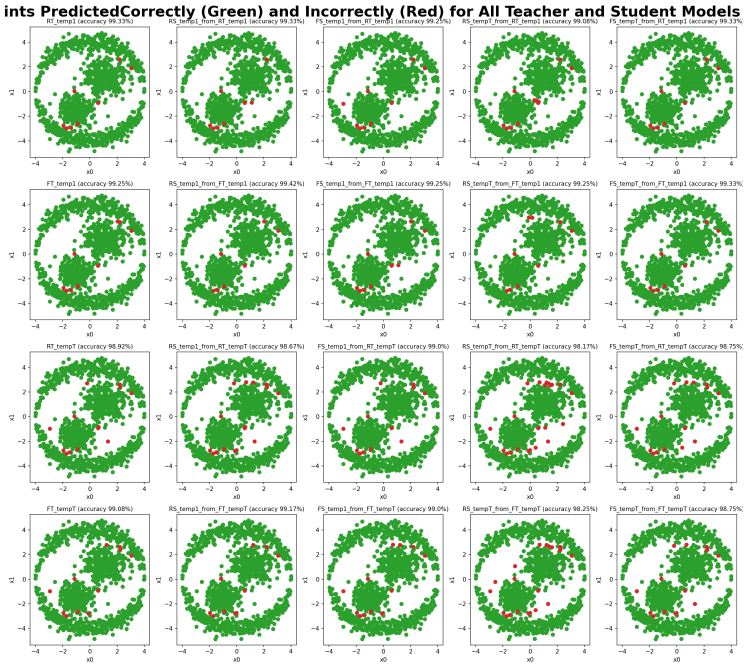

Figure 20: Classification accuracies for all models ($T = 10$).

7.1.2 $T = 40$:

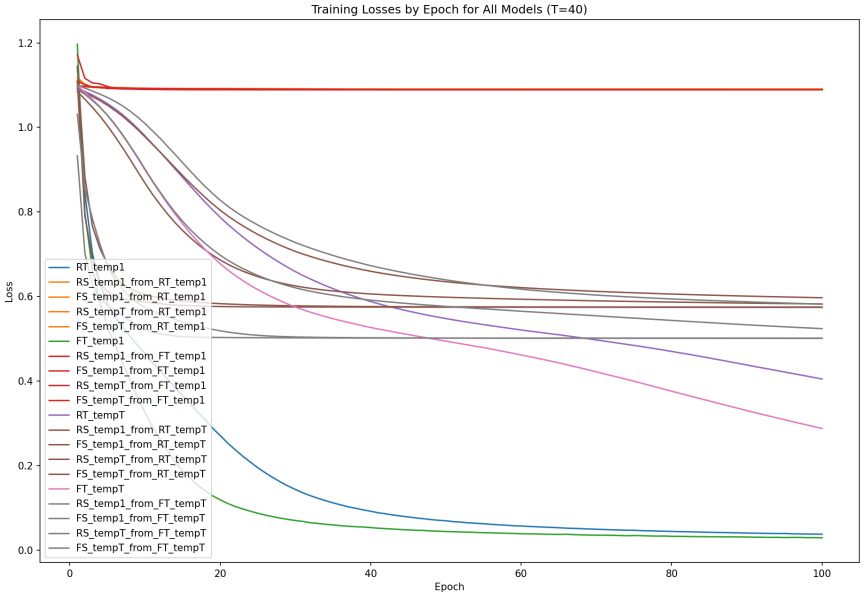

Figure 21: Training loss plots for all models ($T = 40$).

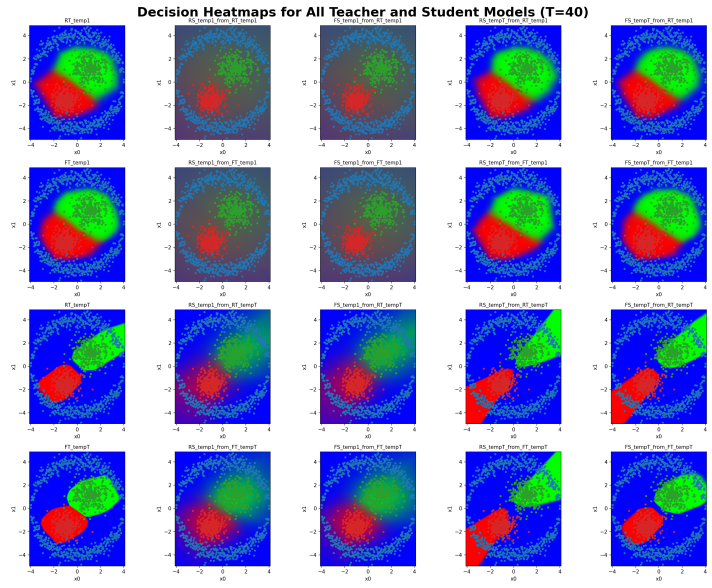

Figure 22: Decision heatmaps for all models ($T = 40$).

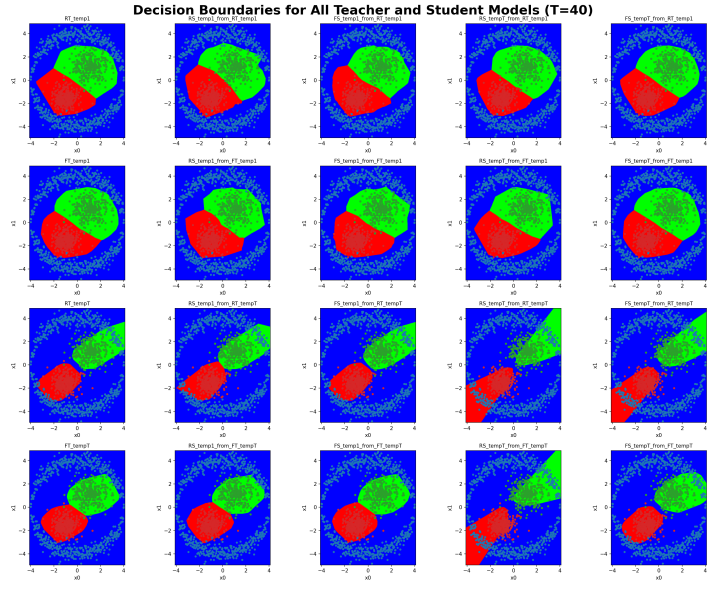

Figure 23: Rounded decision boundaries for all models ($T = 40$).

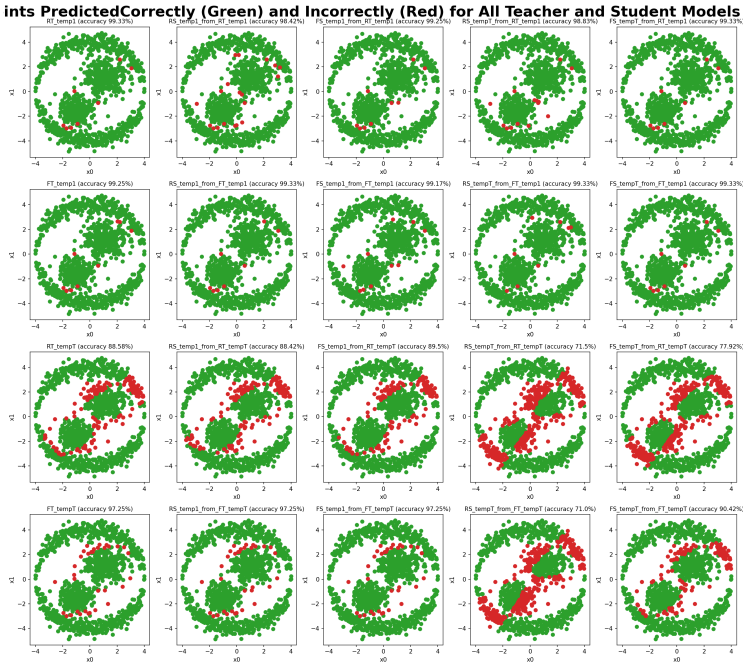

Figure 24: Classification accuracies for all models ($T = 40$).

7.1.3 $T = 70$:

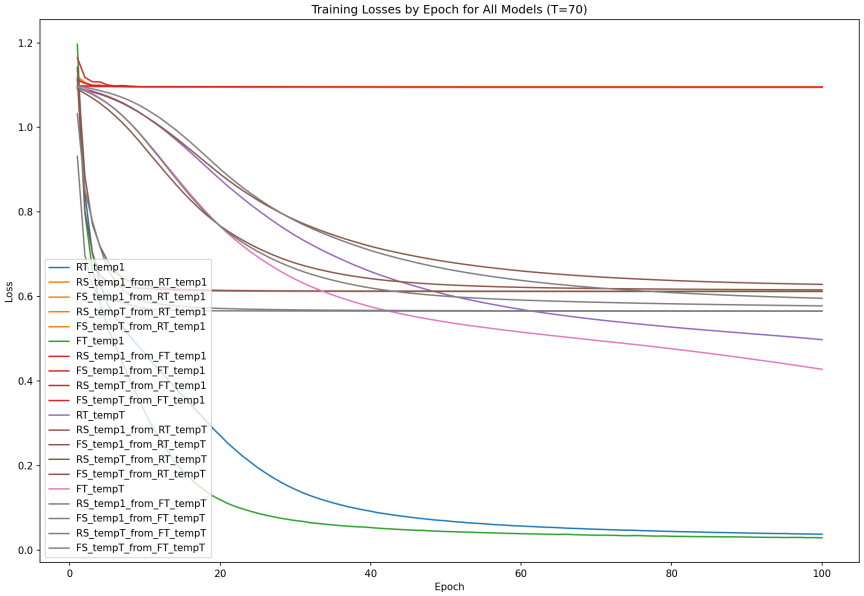

Figure 25: Training loss plots for all models ($T = 70$).

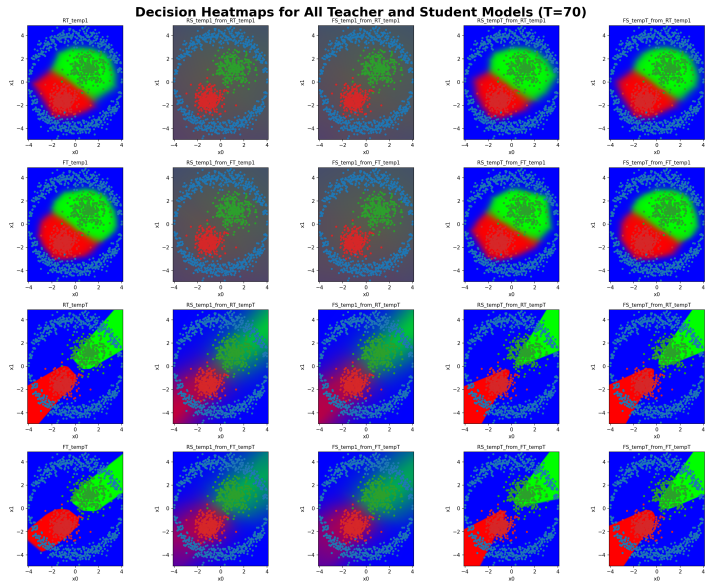

Figure 26: Decision heatmaps for all models ($T = 70$).

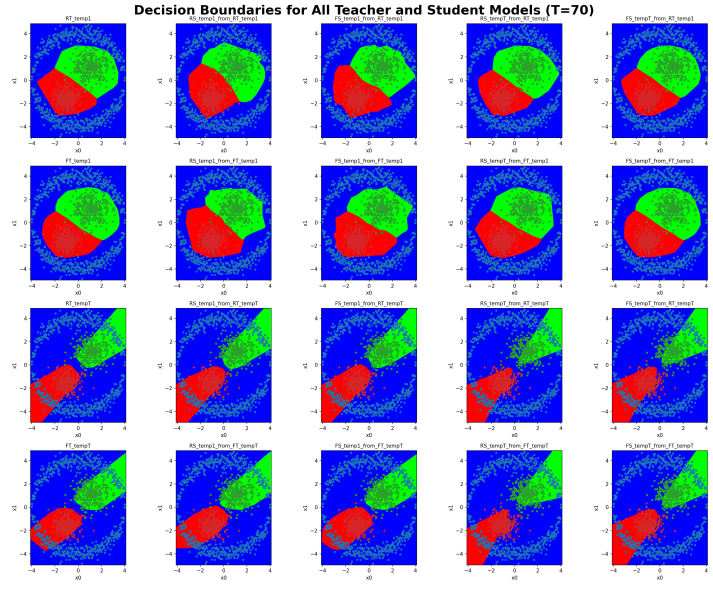

Figure 27: Rounded decision boundaries for all models ($T = 70$).

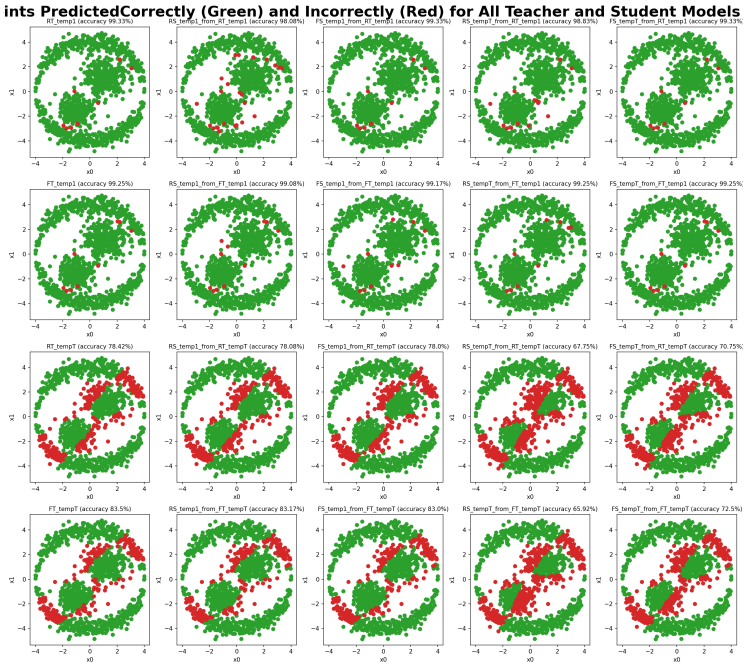

Figure 28: Classification accuracies for all models ($T = 70$).

7.1.4 $T = 100$:

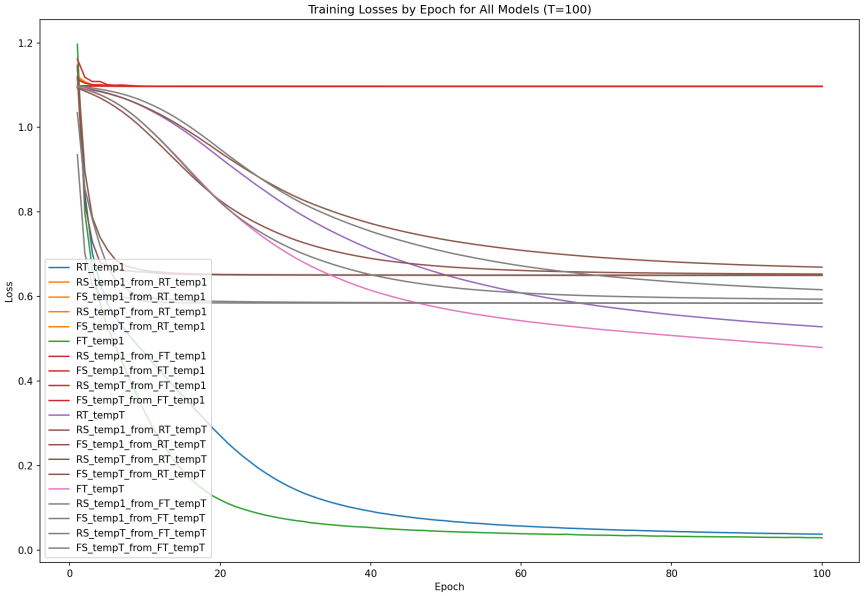

Figure 29: Training loss plots for all models ($T = 100$).

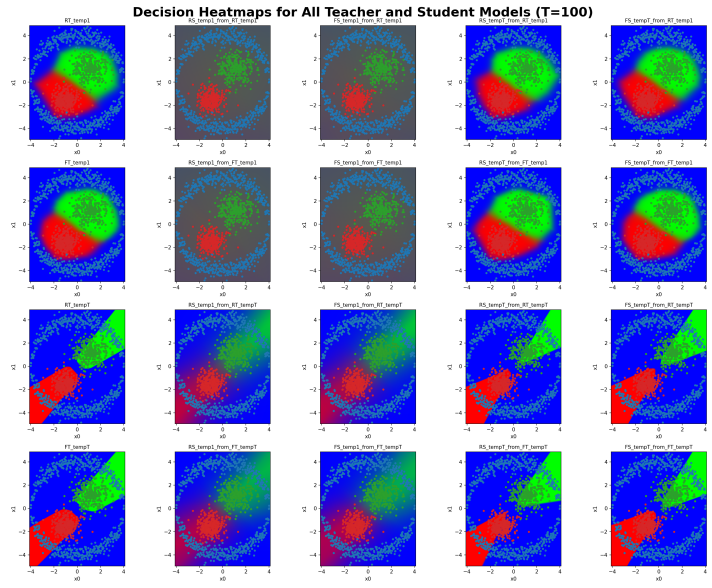

Figure 30: Decision heatmaps for all models ($T = 100$).

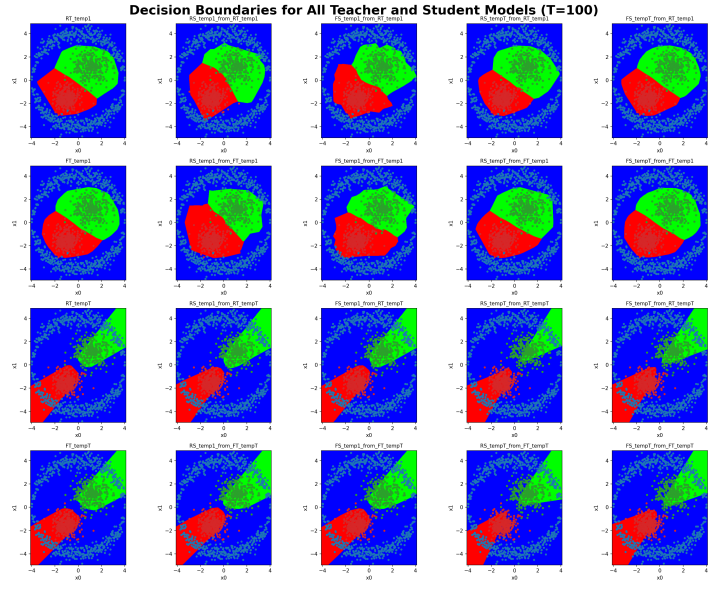

Figure 31: Rounded decision boundaries for all models ($T = 100$).

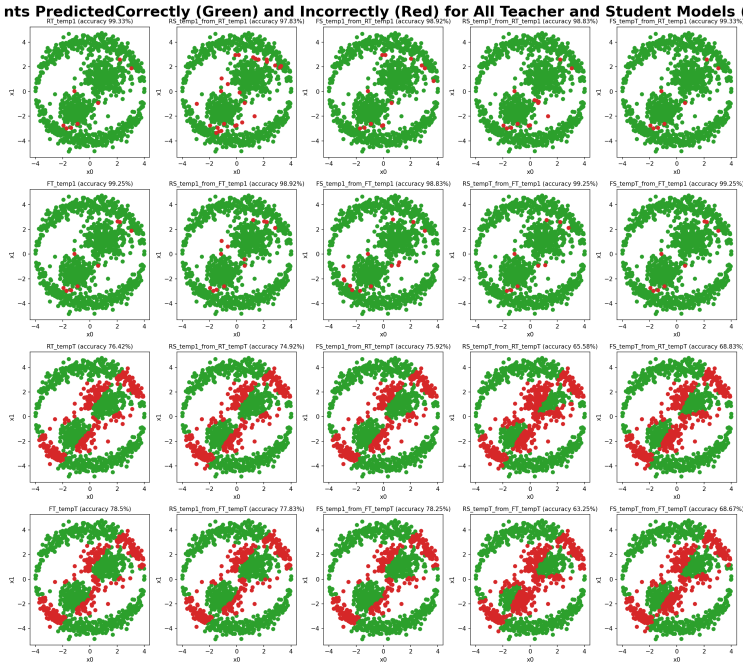

Figure 32: Classification accuracies for all models ($T = 100$).

7.2 CIFAR10 results:

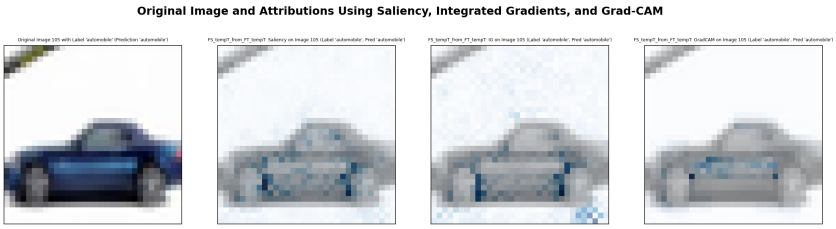

Figure 33: Example of attributions of the defensively distilled model.

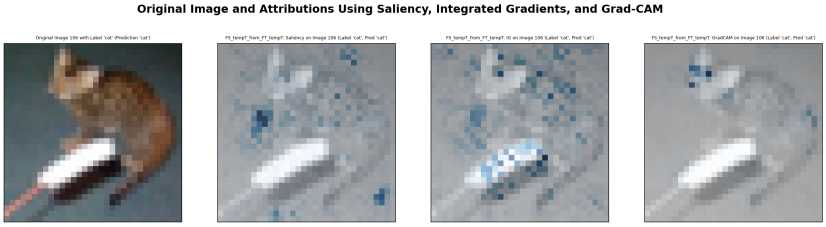

Figure 34: Example of attributions of the defensively distilled model.

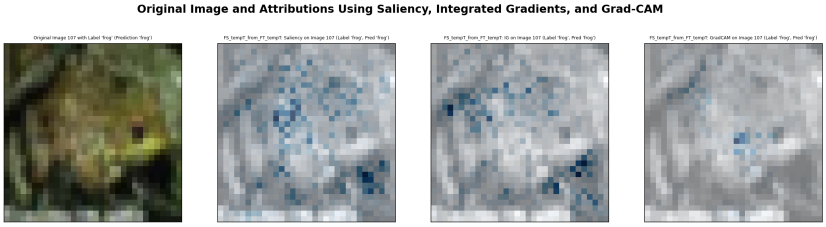

Figure 35: Example of attributions of the defensively distilled model.

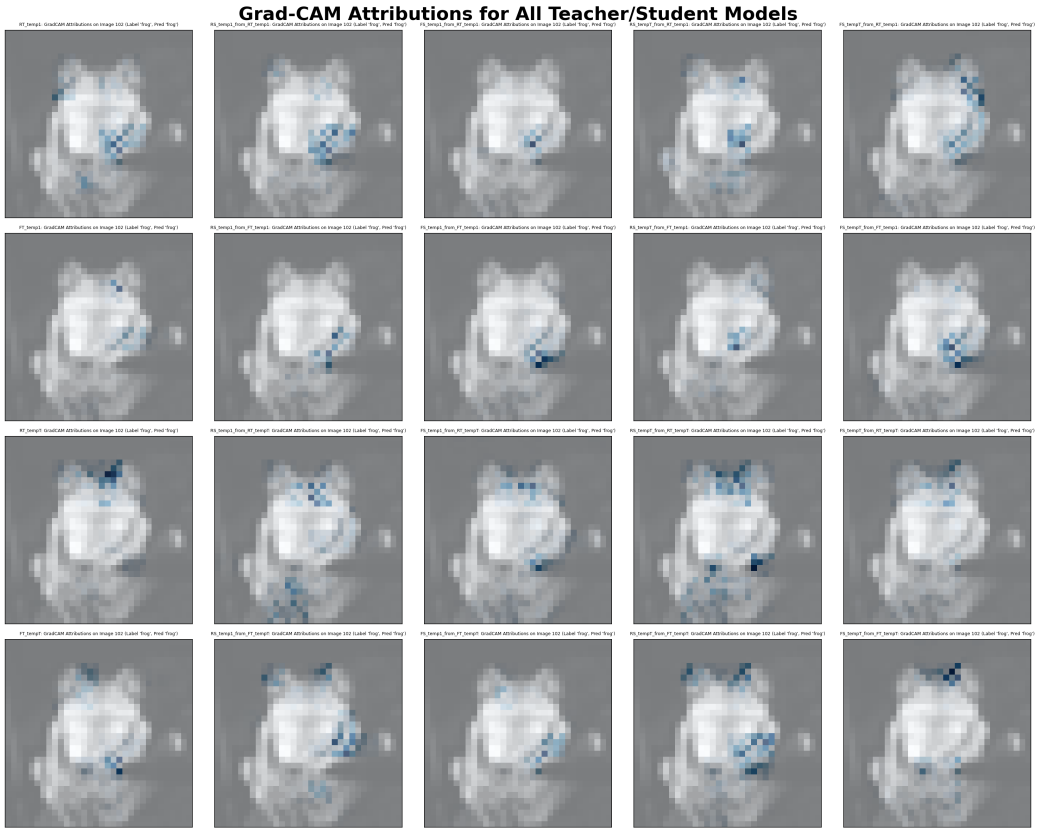

Figure 36: Grad-CAM attributions of each teacher/student for a CIFAR10 image.

7.3 TSRD results:

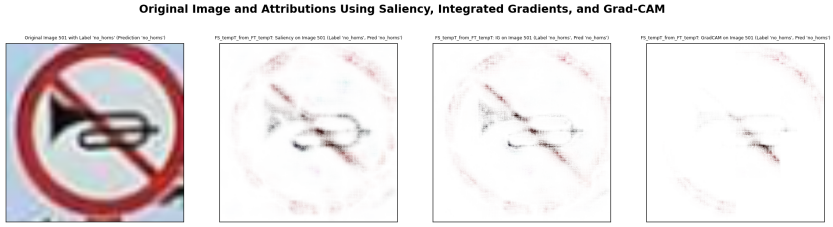

Figure 37: Example of attributions of the defensively distilled model.

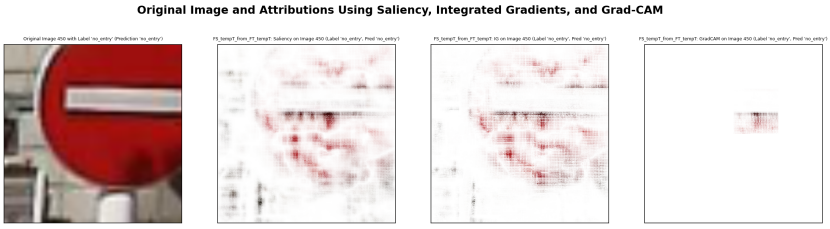

Figure 38: Example of attributions of the defensively distilled model.

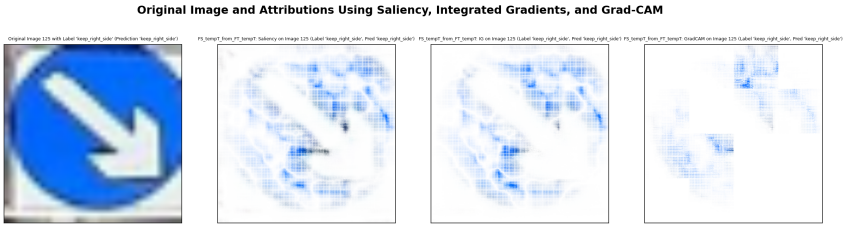

Figure 39: Example of attributions of the defensively distilled model.

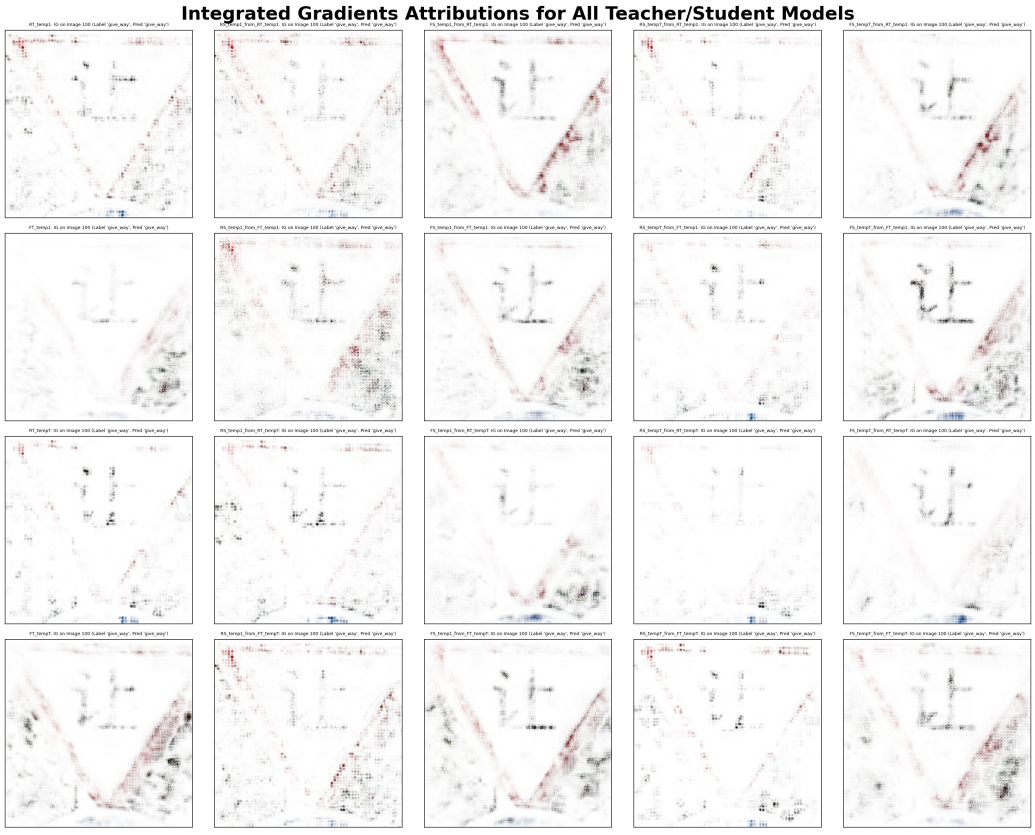

Figure 40: Integrated Gradients attributions of each teacher/student for a TSRD image.