Distillo, ergo sum

Heterogeneous knowledge distillation in medical computer vision

Giacomo Bellini, 1970896, bellini.1970896@studenti.uniroma1.it
Ludovica Mazza, 1917778, mazza.1917778@studenti.uniroma1.it
Matteo Rampolla, 1762214, rampolla.1762214@studenti.uniroma1.it
Daniele Solombrino, 1743111, solombrino.1743111@studenti.uniroma1.it

Abstract

“Distillo, ergo sum” distills [3] a large medical image segmentation Transformer [1] into a small CNN [2], via Heterogeneous Knowledge Distillation (HKD).

Our goals are to formulate hypotheses inspired by class theory, code, comment and verify them, using quantitative and qualitative metrics, like Intersection over Union, Dice score, prediction vs. ground truth visualization, saliency maps and GradCAM.

We hypothesize that HKD helps the CNN capture more global contexts and that distilling the latent embedding spaces is easier than logits matching KD, as a consequence of [1], [5] and [6].

We conclude the report with some possible future directions.

Project setup

Task and dataset

We tested our hypotheses on the “Medical Image Segmentation on Synapse multi-organ CT” challenge, using the “MICCAI 2015 Multi-Atlas Abdomen Labeling Challenge” dataset.
We used an already pre-processed version of the dataset, as in [1].

Models

TransU-Net [1] is our teacher (100M parameters, code available here), while SqueezeU-Net [2] is our student (0.4M, original code ported to PyTorch by us).

Steps

In Step 1, we trained teacher and student from scratch (with joint hyperparameters tuning).
In Step 2, we gathered baseline qualitative and quantitative results.
In Steps 3 and 4, we formulated and verified our first and second hypotheses.

Training

We decided to train both models from scratch, for the following reasons:

1. Finer-grained control over the statistics and data to use in hypothesis verification.
2. Low quality TransU-Net checkpoints distributed by paper [1] authors.
3. Find the same hyperparameters for both models, to have fair comparisons, since different hyperparameters have a huge impact on the loss landscape [4].

We used a 60/20/20 train/validation/test split (2.2k/0.9k/0.9k CT image slices), set a budget of 40 epochs and tracked everything we needed using Weights and Biases

Step 1 - train from scratch

We tested SqueezeU-Net hyperparameters on our data and on TransU-Net (motivation). Training had the desired losses behaviors and we reached performances in Table 1.

Model role	Model name	IoU ↑	Dice score ↑
Teacher	Step 1 TransU-Net	**0.68**	**0.78**
Student	Step 1 SqueezeU-Net	0.42	0.51

Table 1. Quantitative results comparison between TransU-Net and SqueezeU-Net.
Intersection Over Union and Dice score metrics.

Step 2 - baseline qualitative results

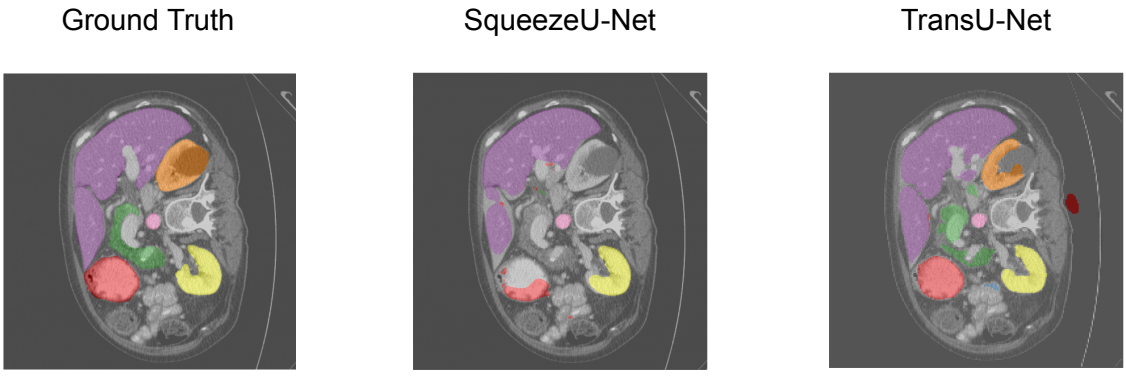

Figure 1. Qualitative results comparison between TransU-Net and SqueezeU-Net.
Ground truth vs. SqueezeU-Net prediction vs. TransU-Net prediction.
TransU-Net gets closer to the ground truth, while SqueezeU-Net struggles across all organs.

TransU-Net is close to the ground truth: all organs are detected and edges are either slightly over-segmented (liver, violet) or under-segmented (left kidney, yellow and stomach, red). The right kidney (orange) is completely under-segmented.

SqueezeU-Net struggles on large organs, since CNNs' inductive bias favors local interactions.

Both models' struggle with right kidney (orange) under-segmentation may be attributed to the organ being represented by two different parts, which have two different color tonalities. TransU-Net's stronger global inductive bias managed to capture at least one of the two parts, while the "split" probably fools SqueezeU-Net locally-focussed inductive bias.

TransU-Net manages to capture the overall shape and size of organs, lacking precision in segmenting edges of organs, which may require additional training epochs or data. SqueezeU-Net tends to under/over-segment most organs.
These considerations are perfectly aligned with models' inductive biases: Transformers tend to shine in capturing local and global contexts, CNNs favor local interactions.

To confirm (or confute) above conclusions, we decided to look at saliency maps [14] and Grad-CAM [15] representations for the liver.

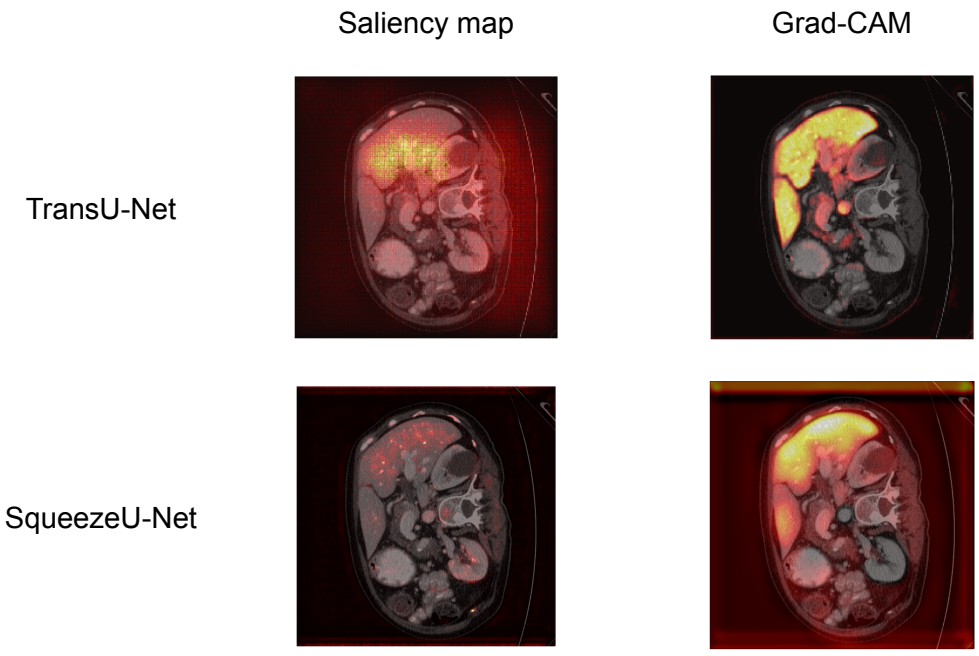

Table 2. Qualitative results comparison between TransU-Net and SqueezeU-Net.
Saliency Map and Grad-CAM.
Both visualizations confirm that both models are learning according to their inductive biases.

In the saliency maps, we can see that TransU-Net focuses on a large-scale portion of the class under analysis, while SqueezeU-Net has a small-scale, patched and sparse focus.

Grad-CAM shows that TransU-Net is once again focussed on a large scale part of the image, while SqueezeU-Net has local concentrations across the entire image.

Both representations confirm our observations, since they show that both models are learning according to their inductive biases.

Step 3 - hypothesis 1

Setup and claim

Due to their inductive biases, CNNs struggle in capturing global contexts, while Transformers can learn local and global interactions both. For these reasons, we hypothesized that distilling a Transformer into a CNN helps the latter gain more global context and we adopted the logits matching KD [3] (Figure 2), using the following loss function:

$$L_{h_1}(X, Y, S, T) = \alpha \cdot CrossEntropyLoss(S(X), Y) + \beta \cdot DiceLoss(S(X), Y) + \delta \cdot KLLoss(S(X), T(X)))$$

- X is the input, Y is the ground truth
- S is the student model (SqueezeU-Net), T is the teacher model (TransU-Net)
- α, β, δ are the weighting factors for the three losses components
- First two components are the image segmentation objective (from TransU-Net paper [1])
- Last component is for the KD via logits matching (from KD paper [3])

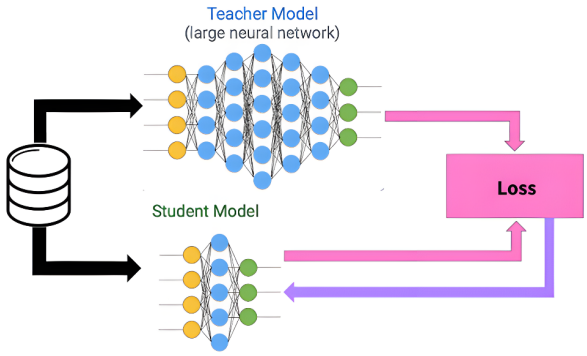

Figure 2. Knowledge distillation via logits matching schema.
Black and pink arrows represent the forward pass. Violet arrow shows backward pass.

Results

Model role	Model name	IoU ↑	Dice score ↑
Student	Step 3 SqueezeU-Net	**0.49**	**0.61**
Student (baseline)	Step 1 SqueezeU-Net	0.42	0.51

Table 3. Quantitative results comparison between Steps 1 and 3 SqueezeU-Net.
KD via logits matching improves SqueezeU-Net quantitative performances.

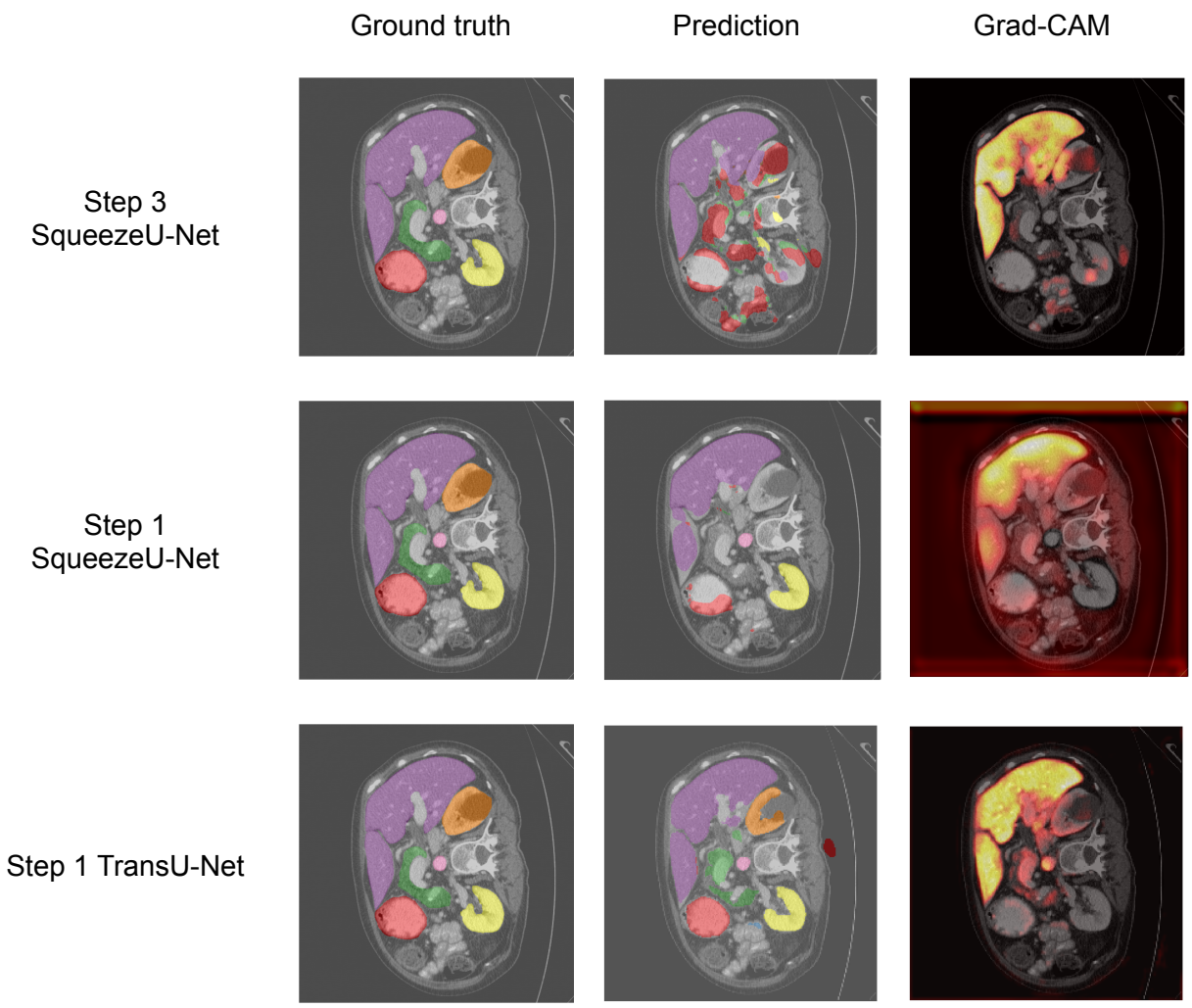

Table 4. Quantitative results comparison between Steps 1 and 3 SqueezeU-Net.
Ground truth prediction and Grad-CAM.
KD via logits matching makes SqueezeU-Net more aware of global contexts, but fails to boost its segmentation metrics.

Step 3 SqueezeU-Net appears to be more aware of the global context, since we can observe that it tries to predict a class for some pixels that were ignored in Step 1.
Most of these new predictions are completely wrong, hinting at the fact that the student is trying to mimic the way the teacher looks at the image, yet it probably needs more training epochs to then assign the correct label to these newly detected pixels.

Grad-CAM visualization perfectly confirms this statement, since Step 3 SqueezeU-Net activations are much much similar to teachers' ones.

Step 4 - hypothesis 2

Setup and claim

[5] shows that train restartings create the same latent spaces in AutoEncoders, up to rotation.
[6] shows that frozen encoder and decoder equipped with small trainable MLPs can be stitched into an AE.
Both papers suggest that, given a task, a dataset and some Neural Networks, their latent spaces tend to follow a very similar structure, which can be easily discovered.

For these reasons, we hypothesize that HKD offers such a learnable transformation and we adopted the KD via latent embedding space matching [7], shown in Figure 3, using this loss:

$$L_{h_2}(X, Y, S, S_e, T_e) = \alpha \cdot CrossEntropyLoss(S(X), Y) + \beta \cdot DiceLoss(S(X), Y) + \delta \cdot L1Loss(S_e(X), T_e(X)))$$

- X is the input, Y is the ground truth (both batched)
- S is the student model (SqueezeU-Net)
- $S_e(X)$ is the latent embedding of the student, $T_e(X)$ is the latent embedding of the teacher
- α, β, δ are the weighting factors for the three losses components
- First two components are the image segmentation objective (from TransU-Net paper [1])
- Last component is for the KD (from [7])

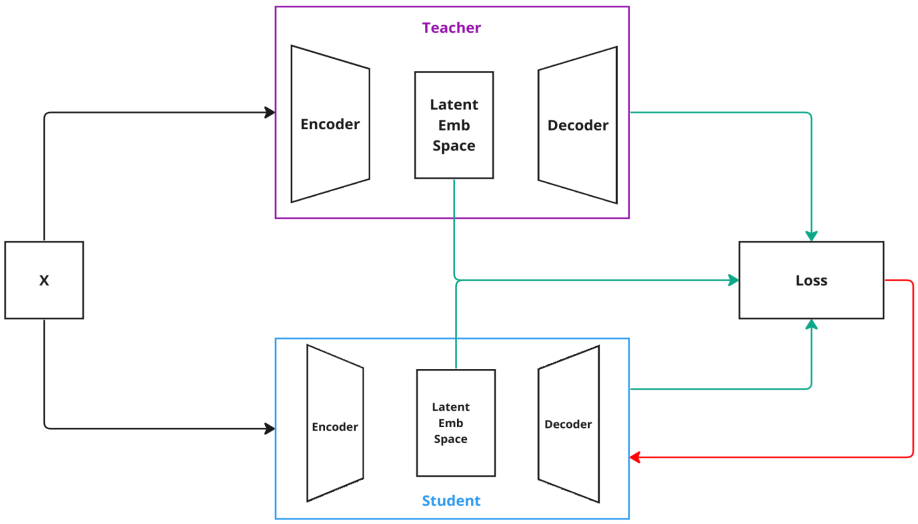

Figure 3. Knowledge distillation via latent embedding space matching schema.
Black and green arrows represent forward pass. Red arrow shows the backward pass.

Results

Step 4 SqueezeU-Net performs worse than Step 1 and 3 students in both quantitative metrics.

Model role	Model name	IoU ↑	Dice score ↑
Student	Step 4 SqueezeU-Net	0.31	0.41
Student (baseline)	Step 3 SqueezeU-Net	**0.49**	**0.61**
Student	Step 1 SqueezeU-Net	0.42	0.51

Table 5. Quantitative metrics comparison between Step 1, 3 and 4 SqueezeU-Net.
KD via latent embedding space matching worsens SqueezeU-Net quantitative performances.

Since quantitative results were not as expected, we skipped qualitative analysis.

Additional claim

We decided instead to focus on the plateau afflicting the latent embedding space loss term.

We attribute the plateau to the low number of parameters in the student (0.4M), which can barely focus on the segmentation task and can not work on the latent space organization too. We tried to increase the number of student trainable parameters up to, 4, 6.7 and 16M. None of them helped, so we concluded that aligning latent spaces of heterogeneous models requires more advanced techniques (e.g. contrastive learning [8]).

Future works

The following directions may be considered to further improve “Distillo, Ergo Sum”

- Latent spaces:
    - Visualizations
    - More advanced alignment techniques
- More advanced HKD techniques, like [10], [11] or [12]
- Switch to homogeneous KD, like [13]

Code

The entire codebase is available in our GitHub repository.

Authors contribution repartition

All group members actively participated in every project step. After key decisions were taken, we split tasks among sub-teams to parallelize work and achieve faster results.

- Giacomo Bellini: Project setup, Step 1, Step 3 and 4 Setup and claim
- Ludovica Mazza: Project setup, Step 2, Step 3 and 4 Results
- Matteo Rampolla: Project setup, Step 2, Step 3 and 4 Results
- Daniele Solombrino: Project setup, Step 1, Step 3 and 4 Setup and claim

Conclusions

Distilling a large medical image segmentation Transformer [1] into a small CNN [2] makes the student more aware of global contexts, but can not boost it to teacher-level performances.

Counter to our original hypothesis, logits matching KD is easier for the student to achieve and performs better than latent embedding space alignment.

Future works include adoption of more advanced techniques related to latent spaces visualizations and alignment, heterogeneous KD or homogeneous KD.

Appendix

Deep Mutual Learning

Before formulating hypothesis 2, we tried to experiment with Deep Mutual Learning [9] (DML), in order to compare it to hypothesis 1.

DML resulted in worse quantitative performances for the teacher and the student both, which actually made sense to us.
In fact, in DML there is no teacher-student paradigm, rather both models learn from each other. In our case, the teacher is much stronger than the student, so, effectively, we are forcing a stronger model to learn from a weak one, which inevitably decreases teacher's performances.

Latent space alignment

After experimenting with hypothesis 2, we performed some ablation studies in which we turned off the segmentation loss terms and tried to optimize the latent space alignment loss term only.

Unfortunately, it did not work very well and basically helped us prove that indeed the task-related loss terms are needed by the small student model in order to get a better understanding of how to properly copy the teacher.

References

1. TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation
2. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters @ ICLR 2017
3. Distilling the Knowledge in a Neural Network @ NIPS 2014
4. Large Scale Structure of Neural Network Loss Landscapes @ NeurIPS 2019
5. Relative representations enable zero-shot latent space communication @ ICLR 2023
6. Revisiting Model Stitching to Compare Neural Representations @ NeurIPS 2021
7. FitNets: Hints for Thin Deep Nets @ ICLR 2015
8. A Simple Framework for Contrastive Learning of Visual Representations @ ICML 2020
9. Deep Mutual Learning @ CVPR 2018
10. Coaching a Teachable Student @ CVPR 2023
11. Heterogeneous Generative KD with Masked Image Modeling @ ECCV 2023
12. Heterogeneous Knowledge Distillation using Information Flow Modeling @ CVPR 2020
13. TinyViT: Fast Pretraining Distillation for Small Vision Transformers @ ECCV 2022
14. Deep Inside CNNs: Visualising Image Classification Models and Saliency Maps
15. Grad-CAM: Visual Explanations of DNNs via Gradient-based Localization @ ICCV 2017