ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Boosting the Lottery Ticket Hypothesis with Knowledge Distillation: Finding Sparser Winning Tickets

Daan Luyckx[1,2,3] and Peter Karsmakers[1,2,3]*†

1- KU Leuven - Dept.of Computer Science, DTAI-ADVISE
Kleinhoefstraat 4, B-2440 Geel - Belgium

2- Leuven.AI - KU Leuven Institute for AI

3- Flanders Make @ KU Leuven

Abstract. The Lottery Ticket Hypothesis (LTH) suggests that dense networks contain sparse, trainable subnetworks, "winning tickets", that can match the original model's performance when trained in isolation. These subnetworks are usually found through iterative pruning, but at high sparsity many potentially effective subnetworks fail to converge under standard training. Knowledge distillation (KD) mitigates this issue by providing richer supervision from a teacher model. We propose the **Knowledge-Distilled Lottery Ticket (KDLT)** procedure, a dual-phase method that applies KD during pruning and retraining to recover stronger sparse subnetworks. Experiments on MNIST, CIFAR-10/100, and Tiny-ImageNet show that KDLT delivers higher accuracy at fixed sparsity or comparable accuracy at higher sparsity.

1 Introduction

Efficient neural network design is critical for deployment on edge hardware, where memory, computational, and energy resources are limited. Although deep learning models have achieved remarkable success, their increasing complexity poses significant challenges for inference in resource-constrained settings. This motivates the development of methods that yield compact models without compromising performance.

The Lottery Ticket Hypothesis (LTH) [1] suggests that dense neural networks contain sparse, trainable subnetworks, or "winning tickets", that can match the task performance (e.g. classification accuracy) of the original model when trained in isolation. Such subnetworks are typically identified through iterative pruning, which removes weights until task performance degrades too excessively. LTH has been explored through both one-shot and iterative pruning strategies [1], with the latter generally achieving superior results at higher sparsity levels. However, achieving dense-model performance at high sparsity is challenging: many subnetworks that are potentially capable simply fail to converge under standard training.

Knowledge distillation (KD) [2] addresses this limitation by using a teacher network to provide richer supervision during training to a smaller student. This

*Corresponding Author: {daan.luyckx, peter.karsmakers}@kuleuven.be

†This work was supported by the Flemish Government's AI Research Program, Flanders Make, and VLAIO through the MEDLI COOCK+ project (HBC.2024.0597).

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

supervision can help recover sparse LTH subnetworks that would otherwise be lost due to poor convergence under vanilla training.

In this work, we introduce the Knowledge-Distilled Lottery Ticket (KDLT), a dual-phase procedure that incorporates KD during both pruning and retraining. We argue that this enables the identification of subnetworks that match or exceed the performance of conventional winning tickets at equal or higher sparsity levels.

The remainder of this paper is organised as follows: Section 2 reviews related work, Section 3 introduces the KDLT algorithm, Section 4 describes the experimental setup, Section 5 presents and discusses the results, and Section 6 concludes with future directions.

2 Related Work

Recent studies have combined KD with the LTH to improve pruning performance, each applying KD at different stages of the pipeline. *Ma et al.* [3] introduce the KD Ticket approach, where a sparse subnetwork is first obtained through standard iterative pruning and then retrained using soft labels from the fully trained dense model. In this setting, KD is applied only during the retraining phase, allowing the pruned network to inherit information from the trained the teacher. In contrast, *Hippocampus et al.* [4] propose KD-LTS, which injects teacher knowledge at initialisation. Instead of pruning by magnitude or gradient criteria, KD-LTS uses responses, features, and relational information obtained from an ensemble of teachers to guide mask selection before any standard training occurs. Other works rely on domain-specific KD after pruning. *Rajaram et al.* [5] combine LTH and KD for recommender systems using a Show-Attend-and-Distill (SAD) mechanism. SAD replaces Kullback-Leibler (KL)-based distillation with an attention-driven feature matching loss, and distillation is used only after pruning or during retraining. Finally, *Li et al.* [6] examine a similar combination in Graph Neural Networks. Their method performs gradual iterative magnitude pruning, and after each pruning step the smaller student network is trained to imitate the output distribution of the pre-pruned teacher. Existing work applies KD either in the pruning or retraining phase.

Our method, KDLT, applies KD in both the pruning and retraining stages. Using conventional soft-label distillation, KDLT transfers teacher knowledge while the pruning mask is being formed, and again when the final sparse subnetwork is trained from its original initialisation. This dual-phase design unifies the benefits of KD-assisted pruning and KD-assisted retraining, and aims to identify smaller winning tickets without the performance degradation found in classical LTH settings.

3 Methodology

In this section, we introduce the proposed KDLT framework, beginning with brief reviews of the LTH [1] and KD [2] before presenting the integrated methodology.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

3.1 The Lottery Ticket Hypothesis

Let $f(x;\theta)$ be a model with initial parameters $\theta_0 \sim D_\theta$. After training for j iterations, the network reaches task performance a. The hypothesis claims that there exists a binary mask $m \in \{0,1\}^{|\theta|}$ such that the subnetwork $f(x; m \odot \theta_0)$ can be trained to performance $a' \geq a$ in $j' \leq j$, while $\|m\|_0 \ll |\theta|$.

Frankle and Carbin identify such subnetworks using iterative magnitude pruning (IMP): (1) initialise $f(x;\theta_0)$; (2) train for j iterations to obtain θ_j; (3) prune the $p\%$ weights with the smallest magnitude to obtain mask m; and (4) reset unpruned weights to their initial values.

IMP can be applied once or repeatedly, with iterative pruning typically yielding smaller and more effective winning tickets [1]. This work, including the proposed KDLT framework, follows the iterative approach.

3.2 Knowledge Distillation

Let z_i denote the logit for class i, and define the temperature-scaled softmax

$$q_i^{(T)} = \frac{e^{z_i/T}}{\sum_j e^{z_j/T}}.$$

Both teacher and student use this softened distribution, yielding $q_t^{(T)}$ and $q_s^{(T)}$. The student minimises a weighted sum of the standard cross-entropy loss with true labels and the KL divergence between softened teacher and student outputs:

$$\mathcal{L}_{\text{total}} = (1-\alpha) \underbrace{\left[-\sum_i y_i \log q_{s,i}^{(1)} \right]}_{\mathcal{L}_{CE}} + \alpha T^2 \underbrace{\left[\sum_i q_{t,i}^{(T)} \log \frac{q_{t,i}^{(T)}}{q_{s,i}^{(T)}} \right]}_{\mathcal{L}_{KD}}. \tag{1}$$

The factor T^2 ensures consistent gradient scaling with softened probabilities.

3.3 The Knowledge-Distilled Lottery Ticket Framework

The KDLT framework combines KD with LTH to identify sparse subnetworks with improved performance at higher sparsity levels. Given a trained teacher network, a student network with the same architecture is then trained via KD, following the self-distillation strategy of Born-Again Networks [7]. After this distillation stage, the student network undergoes global unstructured magnitude pruning. KDLT supports both one-shot and iterative pruning. In the iterative version, a fixed proportion of the lowest-magnitude weights is pruned at each step, and surviving weights are reset to their initial values. After pruning, the subnetwork is retrained using the $\mathcal{L}_{\text{total}}$ with the same teacher model, which remains fixed throughout the procedure. By integrating KD into both pruning and retraining phases, KDLT is expected to improve identifying winning tickets with increased sparsity. A formal description of the method is provided in Algorithm 1.

If $K = 1$, the algorithm performs one-shot pruning. If $K > 1$, pruning is iterative. In all cases, the teacher remains fixed, and the student is retrained using KD after pruning.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

Algorithm 1 Knowledge-Distilled Lottery Ticket (KDLT)

Require: Pretrained teacher model $\theta_{teacher}$, training data $\mathcal{D}$, model $f(x;\theta)$, pruning rate p, nr. of iterations K

1: Initialize weights $\theta_0 \sim D_\theta$ of student model with same architecture as teacher
2: Train student model using KD to obtain $\theta_{student}$
3: **for** $k = 1$ to K **do**
4: Create mask m to prune the $p\%$ weights with the smallest absolute value.
5: Reset surviving weights to values in θ_0
6: Retrain $f(x; m \odot \theta_0)$ using $\mathcal{L}_{\text{total}}$
7: **end for**
8: **return** Sparse student model $f(x; m \odot \theta_0)$

4 Experiments

To evaluate our approach, we conducted experiments on four standard image-classification benchmarks. CIFAR-10/100 consists of small images across 10 and 100 classes, respectively [8], while MNIST contains greyscale images of handwritten digits [9]. Detailed specifications for these datasets can be found in their original references. Tiny-ImageNet, a 200-class subset of ImageNet, provides 64×64 images with 100,000 training and 10,000 validation/test samples (500 per class) [10, 11]. For CIFAR and Tiny-ImageNet, we employ a ResNet-18 backbone [12], and for MNIST we use the LeNet-5 [13] architecture. All datasets use an 80/20 split of the official training set for training/validation to ensure a consistent evaluation protocol. For training, all experiments are implemented in PyTorch, and we apply standard data augmentation following [13, 12]. For TinyImageNet, we additionally use *RandAugment* [14]. All networks are initialised using uniform *Kaiming He* initialisation [15]. For all KD experiments, the temperature T and weighting factor α are set to 2 and 0.25, respectively, selected via a grid search on ResNet-18 with CIFAR-10. All remaining training configurations and hyperparameters are summarised in Table 1.

Dataset	**Network**	**Batch Size**	**Epochs**	**Optimizer Settings**	
				SGD	**Adam**
MNIST	LeNet-5	128	250	$lr = 0.1$ $\beta = 0.9$ weight decay $= 5 \times 10^{-4}$	$lr = 0.001$
CIFAR-10	ResNet-18				
CIFAR-100					
Tiny-ImageNet		256			

Note:Applied LR scheduling (halve-on-plateau, patience=20) and early stopping (patience=100).

Table 1: Training settings across datasets, networks, and optimizers.

5 Results and Discussion

Figure 1 summarises the experimental results, comparing the unpruned baseline, the iterative LTH method [1], and our proposed iterative KDLT approach. Results are shown for both Adam [16] and SGD [17] optimizers. Each pruning method is run for 25 iterations, removing 20% of the remaining weights per step. The reported sparsity levels therefore reflect the discrete sparsity reached after each iteration rather than a uniformly spaced linear scale.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

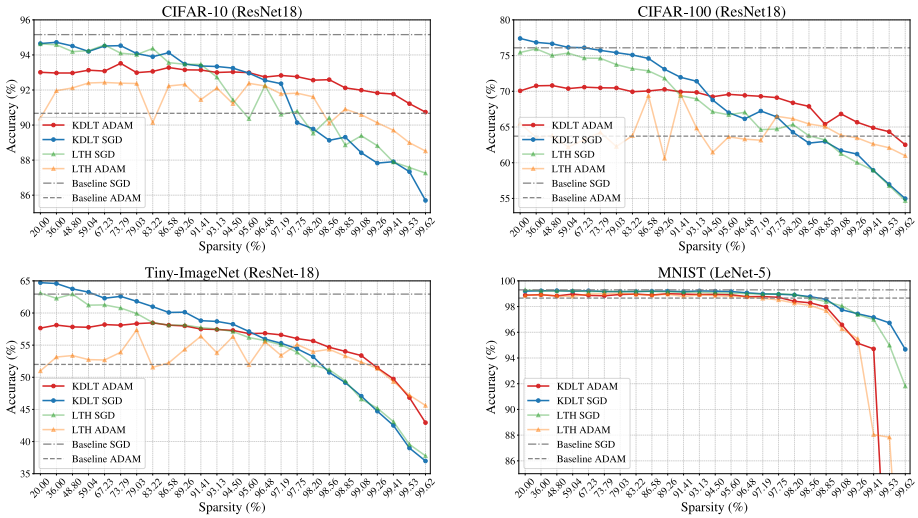

Fig. 1: Accuracy-Sparsity Curves Comparing the Baseline, LTH, and KDLT Pruning Methods Across All Datasets

KDLT generally outperforms the standard LTH approach across datasets. On MNIST, all methods achieve similarly high accuracy due to the simplicity of the task, though KDLT-SGD shows an edge at extreme sparsity. On CIFAR-10/100, KDLT-Adam shows an advantage over LTH-Adam as sparsity increases, whereas KDLT-SGD offers only moderate or no improvements over LTH-SGD. The SGD-based methods lead at lower sparsity levels, while the Adam-based methods prove more resilient at higher sparsity. For Tiny-ImageNet, KDLT-Adam demonstrates modest gains over LTH-Adam at low-to-mid sparsity, and the same crossover behavior is observed. These results indicate that KDLT effectively transfers knowledge from the dense model and stabilizes sparse training, particularly in more challenging settings.

Optimizer choice thus plays a significant role. While trends vary by dataset and sparsity level, the results indicate that the interaction between optimizer and pruning intensity consistently affects final performance and should be considered when designing sparse training pipelines.

6 Conclusion

We introduced the Knowledge-Distilled Lottery Ticket (KDLT) algorithm, which integrates knowledge distillation into both the pruning and retraining stages of the lottery ticket framework. Experimental results on MNIST, CIFAR-10/100, and Tiny-ImageNet (using LeNet-5 and ResNet-18 models) demonstrate that KDLT produces sparser winning tickets with higher test accuracy than standard iterative LTH. These findings confirm the empirical advantage of combining knowledge distillation with the lottery ticket approach: the combined method

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

yields compact, high-performing networks that maintain accuracy even under aggressive pruning.

Future Work will examine how the KD hyperparameters α and T, influence the quality of sparse subnetworks. In addition, expanding the KDLT framework to a broader range of architectures and datasets will help assess its generality beyond the settings explored here. Finally, a deeper investigation into optimizer behaviour under sparsity is warranted, including the exploration of hybrid or more advanced strategies to better understand and potentially mitigate the differing degradation patterns observed across optimizers.

References

[1] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks, 2019.

[2] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015.

[3] Haoxiang Ma, Tianlong Chen, Tao-Kun Hu, Chaojian You, Xueying Xie, and Zhangyang Wang. Good students play big lottery better. *arXiv preprint arXiv:2101.03255*, 2021.

[4] David S. Hippocampus. Winning the lottery once and for all: Towards pruning neural networks at initialization. *Submitted to Transactions on Machine Learning Research*, 2024. Rejected.

[5] Rajaram R, Manoj Bharadhwaj, Vasan VS, and Nargis Pervin. Enhancing scalability in recommender systems through lottery ticket hypothesis and knowledge distillation-based neural network pruning, 2024.

[6] Qiang Li, Xingyi Tan, Yonghao Tan, and Zhijian Xu. Joint gradual pruning and knowledge distillation for identifying graph lottery tickets. *Journal of Intelligent & Fuzzy Systems*, 49(5):1137–1149, 2025.

[7] Tommaso Furlanello, Zachary C. Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks, 2018.

[8] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Canada, 2009. Technical Report.

[9] Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. The mnist database of handwritten digits. `http://yann.lecun.com/exdb/mnist/`.

[10] Ya Le and Xuan S. Yang. Tiny imagenet visual recognition challenge. `https://tinyimagenet.cs231n.stanford.edu/`, 2015. Stanford CS231N.

[11] Stanford CS231N. Tiny imagenet challenge. `https://tinyimagenet.cs231n.stanford.edu/`, 2015.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[13] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pages 2278–2324, 1998.

[14] Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPR*, 2020.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, 2015.

[16] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Optimization. In *Deep Learning*, chapter 8. MIT Press, 2016.