MagMax: Leveraging Model Merging for Seamless Continual Learning

Daniel Marczak*[1,2], Bartłomiej Twardowski[1,5,6], Tomasz Trzciński[1,2,4], and Sebastian Cygert[1,3]

[1] IDEAS NCBR
[2] Warsaw University of Technology
[3] Gdańsk University of Technology
[4] Tooploox
[5] Autonomous University of Barcelona
[6] Computer Vision Center

Abstract. This paper introduces a continual learning approach named MagMax, which utilizes model merging to enable large pre-trained models to continuously learn from new data without forgetting previous knowledge. Traditional continual learning methods aim to reduce forgetting *during* task training. MagMax, on the other hand, combines sequential fine-tuning with a maximum magnitude weight selection for effective knowledge integration *after* training on a new task outperforming traditional CL methods.

1 Introduction

Large pre-trained models allow unprecedented performance improvements across many challenging tasks [1, 2, 9, 19, 24, 29]. To keep up with the ever-changing world, these models should adapt continuously and assimilate knowledge from the stream of new data, which is the objective of Continual Learning (CL) [13,15,22]. Traditionally, CL approaches used regularization to retain the knowledge from previous tasks [10, 14], grow the network while learning new tasks [20, 28], or use a replay buffer to limit the catastrophic forgetting [6, 25, 30]. However, model merging emerged as a new paradigm of adapting pre-trained models. It allows to consolidate the knowledge of multiple independently fine-tuned task-specific models into one multi-task model without any additional training. Various methods base on selecting or interpolating the weights of task-specific models [7, 16, 18, 21, 27]. Contrary to the traditional CL methods, which focus on alleviating forgetting *during* training on new tasks, model merging allows to seamlessly consolidate the knowledge *after* the training on new tasks leaving the training procedure unchanged. Inspired by a recent progress in model merging, we propose MagMax, a novel method for continual learning that utilizes sequential fine-tuning and model merging via maximum magnitude selection (see Fig. 3).

* Corresponding author, email: daniel.marczak.dokt@pw.edu.pl

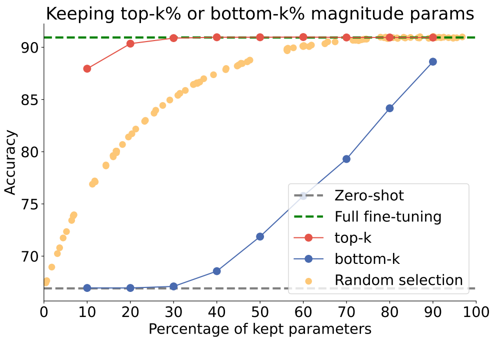

Fig. 1: Only a small fraction of parameters that changed the most during fine-tuning is responsible for improved performance.

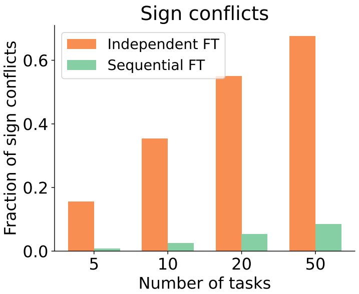

Fig. 2: Sequential fine-tuning encourages consistent directions of parameter updates. We report sign conflicts after trimming 80% of the lowest magnitude params in each task vector.

2 Background and motivation

Problem setting. We consider a problem of continual learning of large pre-trained models. We assume access to a pre-trained model parametrized by d weights $\theta_0 \in \mathbb{R}^d$. Our goal is to adapt the model to a sequence of disjoint tasks $\{D_1, D_2, \ldots, D_n\}$ one task at a time. We investigate *exemplar-free* scenario. We consider two fine-tuning scenarios: (1) independent (Ind FT) - starts from pre-trained weights θ_0, and (2) sequential (Seq FT) - starts from the weights of the model fine-tuned on the sequence of previous tasks, *i.e.* when fine-tuning on task D_t, we start from θ_{t-1} which was trained on $\{D_1, D_2, \ldots, D_{t-1}\}$. We use a notion of task vector [7] that is an element-wise difference between the fine-tuned model and the pre-trained model, *i.e.* $\tau_i = \theta_i - \theta_0$.

Motivation. We motivate our method with the two following hypotheses.

$\mathcal{H}1$***: Parameters that change the most during fine-tuning are the most important for the task.*** We fine-tune a model on and create a task vector τ. Then, we keep only k% of parameters that are selected at random, or according to their magnitude (lowest or highest) and remove the rest. Finally, we apply the pruned task vector to the pre-trained model and evaluate its performance (Fig. 1). We observe that only a small fraction of high-magnitude parameters in task vectors are relevant for the model performance what validates $\mathcal{H}1$.

$\mathcal{H}2$***: Sequential fine-tuning reduces sign conflicts.*** When fine-tuning the model on several tasks, we can observe a disagreement between the directions of task-specific updates. Such a situation is denoted as *sign conflict* [27] results in interference between tasks when merging models, and hence reduced performance of the final model. In this work, we postulate that sequential fine-tuning can reduce the number of sign conflicts. To verify this hypothesis, we use Ind Ft and Seq FT and count the conflicts of top-20% parameters in corresponding task vectors (Fig. 2). We observe that sequential fine-tuning significantly reduces the sign conflicts validating $\mathcal{H}2$.

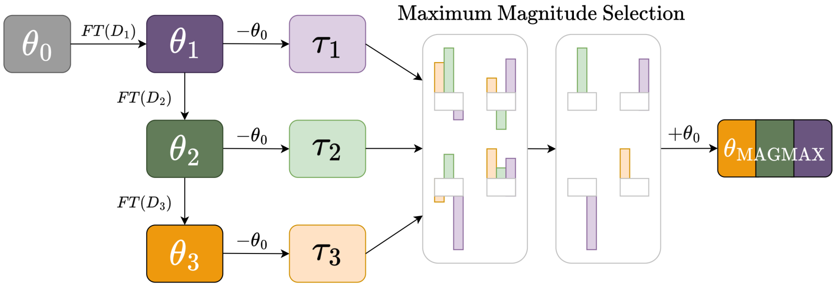

Fig. 3: Overview of the proposed MagMax method for continual learning. We sequentially fine-tune the model on the subsequent tasks and create task vectors τ_i by subtracting the weights of the pre-trained model θ_0. Then we merge the task vectors using MagMax strategy which selects the parameters of task vectors by highest magnitude. Finally, we apply merged task vector to the pre-trained model to obtain a multitask model θ_{MagMax}.

3 MagMax

Based on the motivations introduced in the previous Section, we introduce MagMax. It is a novel method for continual learning that utilizes sequential fine-tuning, following $\mathcal{H}2$, and model merging based on selecting the parameters of the highest magnitude, following $\mathcal{H}1$. Given a new task, D_t, our method consists of two steps: **(1) Sequential adaptation:** We obtain the new weights of the model θ_t by fine-tuning it on D_t. Importantly, we start from the weights of the model fine-tuned on previous tasks θ_{t-1}. **(2) Knowledge consolidation:** We consolidate task-specific knowledge using model merging. Firstly, we create task vectors for all tasks seen so far: $\{\tau_i\}_{i=1}^{t}$, where $\tau_i = \theta_i - \theta_0$. Then, for each parameter $p \in \{1, 2, \dots, d\}$, we select the value τ^p_{MagMax} by the maximum magnitude out of all the task vectors. Lastly, we apply the resulting task vector τ_{MagMax} to the pre-trained model $\theta_{\text{MagMax}} = \theta_0 + \lambda * \tau_{\text{MagMax}}$, where λ is a scaling factor.

4 Experimental setup

We use CIFAR100 [12], ImageNet-R [5], CUB200 [23] and Cars [11] splitted into into N equal subsets of disjoint classes, where $N \in \{5, 10, 20, 50\}$. We use CLIP pre-trained model [19] with ViT/B-16 [3] image encoder. We follow the training procedure from [8]. We train CIFAR100 and ImageNet-R for 10 epochs each task, and CUB200 and Cars for 30 epochs. We use the final classification layer output by CLIP's text encoder and keep it frozen during fine-tuning, following [8]. We compare MagMax against CL baselines **LwF** [14] and **EWC** [10] as well as recent model merging strategies, Model Soup (**Avg**) [26], Task Arithmetic (**TA**) [7] and TIES-Merging (**TIES**) [27]. Additionally, we introduce a simple baseline dubbed **RandMix** which randomly selects each parameter from one of the fine-tuned

Table 1: MagMax outperforms continual learning methods and merging-based approaches on a wide variety of scenarios (acc (%) after the final task).

	CIFAR100				ImageNet-R				CUB200			Cars			Avg
Method	/5	/10	/20	/50	/5	/10	/20	/50	/5	/10	/20	/5	/10	/20	
Zero-shot	66.91				77.73				56.08			64.71			67.21
Joint	90.94				87.55				81.57			88.21			87.38
LwF	83.25	73.45	72.05	68.84	81.15	82.97	81.82	80.32	**65.12**	60.67	**58.90**	71.72	**69.84**	62.98	72.36
EWC	**84.41**	76.24	75.39	72.97	82.15	82.42	81.48	81.47	59.10	54.49	53.31	69.46	60.78	57.42	70.79
RandMix	81.55	77.04	75.36	72.91	83.10	81.88	80.18	78.50	59.86	58.53	58.08	67.32	65.62	64.95	71.78
MaxAbs	81.95	76.75	74.39	73.04	83.03	82.33	80.92	79.33	60.15	58.01	56.59	67.36	63.55	58.95	71.17
Avg	81.41	77.04	75.29	72.92	83.08	81.87	80.27	78.53	59.77	58.44	58.01	67.37	65.59	64.88	71.75
TIES	81.72	77.23	74.66	73.76	83.08	82.27	80.83	79.57	60.94	58.22	56.97	70.45	64.90	61.17	71.84
MagMax	84.16	**80.41**	**78.49**	**76.75**	**83.60**	**83.33**	**82.27**	**81.75**	63.89	**60.74**	**58.90**	**73.61**	69.28	**65.84**	**74.50**

Table 2: Knowledge consolidation step from MagMax improves the performance of regularization-based CL methods but does not outperform vanilla MagMax.

	CIFAR100				ImageNet-R				CUB200			Cars			Avg
Method	/5	/10	/20	/50	/5	/10	/20	/50	/5	/10	/20	/5	/10	/20	
LwF	83.25	73.45	72.05	68.84	81.15	82.97	81.82	80.32	65.12	60.67	58.89	71.72	69.84	62.98	72.36
LwF + MagMax	82.68	77.61	75.81	72.65	82.55	82.52	81.98	80.63	64.53	61.17	59.60	73.29	71.04	67.85	73.85
Δ	-0.57	+4.16	+3.76	+3.81	+1.40	-0.45	+0.16	+0.31	-0.59	+0.50	+0.71	+1.57	+1.20	+4.87	+1.49
EWC	84.41	76.24	75.39	72.97	82.15	82.42	81.48	81.47	59.10	54.49	53.31	69.46	60.78	57.42	70.79
EWC + MagMax	82.34	77.73	77.66	77.03	82.07	83.02	82.35	81.60	63.57	60.61	59.15	72.83	69.59	66.00	73.97
Δ	-2.07	+1.49	+2.27	+4.06	-0.08	+0.60	+0.87	+0.13	+4.47	+6.12	+5.84	+3.37	+8.81	+8.58	+3.18
MagMax	84.16	80.41	78.49	76.75	83.60	83.33	82.27	81.75	63.89	60.74	58.90	73.61	69.28	65.84	74.50

models, *i.e.* $\theta_m^p \sim \{\theta_i^p\}_{i=1}^N$. We also evaluate **MaxAbs** baseline, which is basically MagMax with Ind Ft instead of Seq. Finally, we present **zero-shot** (pre-trained model), and **joint** (model fine-tuned on the whole dataset).

5 Main results

Class-incremental learning. Tab. 1 presents the comparison of MagMax with CL methods and merging-based baselines on various class-incremental learning benchmarks. MagMax consistently outperforms the competitors across the scenarios, achieving on average 2.1% better results than the second best method. Interestingly, simple baselines that merge independent fine-tunings by averaging (Avg) or even randomly mixing (RandMix) the weights, are close competitors to CL methods and other merging strategies.

Does model merging help CL methods? We investigate if knowledge consolidation via model merging helps to improve the performance of CL methods. We modify MagMax and instead of performing Seq FT, we train the model using one of the regularization-based CL methods (Tab. 2). We observe that adding model merging significantly improves the performance of LwF and EWC in almost every scenario. Interestingly, neither of these combinations significantly outperform MagMax which uses naive Seq FT, traditionally known for causing catastrophic forgetting [4,17]. These results show that model merging is a promising technique for consolidating the knowledge *after* the training instead of *during* the training.

Selecting high magnitude parameters promotes consistent update directions. In this Section we set and verify the following hypothesis: *parameters which update directions were consistent across tasks tend to have higher magnitude.* We define an update direction as a sign of parameter change when trained on a given task, $\text{sgn}(\Delta\theta_t^p) = \text{sgn}(\theta_t^p - \theta_{t-1}^p)$. For each parameter in each sequentially fine-tuned task vector, we calculate the number of consistent update directions n. Fig. 4 presents the relation of magnitude of task vectors' parameters and the consistency of update directions. We observe that the parameters with higher consistency tend to have higher magnitude. Therefore, we can think of maximum magnitude selection as a proxy for selecting the updates that multiple tasks agree on.

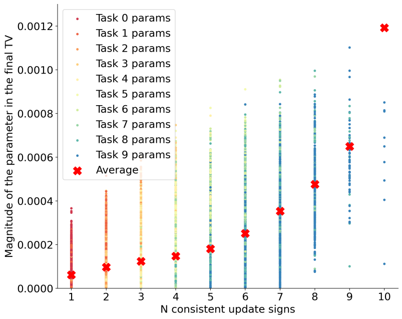

Fig. 4: Magnitude of task vectors' parameters are correlated with the consistency of the update direction in the subsequent tasks.

Acknowledgments

Daniel Marczak is supported by National Centre of Science (NCN, Poland) Grant No. 2021/43/O/ST6/02482. This research was partially funded by National Science Centre, Poland, grant no: 2020/39/B/ST6/01511, 2022/45/B/ST6/02817 and 2023/51/D/ST6/02846. Bartłomiej Twardowski acknowledges the grant RYC2021-032765-I. This paper has been supported by the Horizon Europe Programme (HORIZON-CL4-2022-HUMAN-02) under the project "ELIAS: European Lighthouse of AI for Sustainability", GA no. 101120237. We gratefully acknowledge Polish high-performance computing infrastructure PLGrid (HPC Center: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2023/016393.

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. ECCV (2020) 1
2. Caron, M., Touvron, H., Misra, I., Jegou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. ICCV (2021) 1
3. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021) 3
4. French, R.M.: Catastrophic forgetting in connectionist networks. Trends in Cognitive Sciences (1999) 4

5. Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T.L., Parajuli, S., Guo, M., Song, D., Steinhardt, J., Gilmer, J.: The many faces of robustness: A critical analysis of out-of-distribution generalization. ICCV (2020) 3
6. Hou, S., Pan, X., Loy, C.C., Wang, Z., Lin, D.: Learning a unified classifier incrementally via rebalancing. In: CVPR (2019) 1
7. Ilharco, G., Ribeiro, M.T., Wortsman, M., Schmidt, L., Hajishirzi, H., Farhadi, A.: Editing models with task arithmetic. In: ICLR (2023) 1, 2, 3
8. Ilharco, G., Wortsman, M., Gadre, S.Y., Song, S., Hajishirzi, H., Kornblith, S., Farhadi, A., Schmidt, L.: Patching open-vocabulary models by interpolating weights. In: NeurIPS (2022) 3
9. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. ICCV (2023) 1
10. Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al.: Overcoming catastrophic forgetting in neural networks. PNAS (2017) 1, 3
11. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3D Object representations for fine-grained categorization. In: ICCV Workshops (2013) 3
12. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. University of Toronto (2009) 3
13. Lange, M.D., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., Tuytelaars, T.: A continual learning survey: Defying forgetting in classification tasks. IEEE TPAMI (2019) 1
14. Li, Z., Hoiem, D.: Learning without forgetting. IEEE TPAMI (2018) 1, 3
15. Masana, M., Liu, X., Twardowski, B., Menta, M., Bagdanov, A.D., van de Weijer, J.: Class-incremental learning: Survey and perfoxrmance evaluation on image classification. IEEE TPAMI (2023) 1
16. Matena, M., Raffel, C.: Merging models with fisher-weighted averaging. In: NeurIPS (2021) 1
17. McCloskey, M., Cohen, N.J.: Catastrophic interference in connectionist networks: The sequential learning problem. In: Psychology of Learning and Motivation (1989) 4
18. Ortiz-Jiménez, G., Favero, A., Frossard, P.: Task arithmetic in the tangent space: Improved editing of pre-trained models. In: NeurIPS (2023) 1
19. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML (2021) 1, 3
20. Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R.: Progressive neural networks. CoRR (2016) 1
21. Singh, S.P., Jaggi, M.: Model fusion via optimal transport. In: NeurIPS (2020) 1
22. van de Ven, G., Tuytelaars, T., Tolias, A.: Three types of incremental learning. Nature Machine Intelligence (2022) 1
23. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011) 3
24. Wang, C.Y., Bochkovskiy, A., Liao, H.: Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. CVPR (2022) 1
25. Wang, F., Zhou, D., Ye, H., Zhan, D.: FOSTER: feature boosting and compression for class-incremental learning. In: ECCV (2022) 1

26. Wortsman, M., Ilharco, G., Gadre, S.Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A.S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., et al.: Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In: ICML (2022) 3
27. Yadav, P., Tam, D., Choshen, L., Raffel, C., Bansal, M.: TIES-merging: Resolving interference when merging models. In: NeurIPS (2023) 1, 2, 3
28. Yan, S., Xie, J., He, X.: DER: Dynamically expandable representation for class incremental learning. In: CVPR (2021) 1
29. Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training. ICCV (2023) 1
30. Zhao, B., Xiao, X., Gan, G., Zhang, B., Xia, S.: Maintaining discrimination and fairness in class incremental learning. In: CVPR (2020) 1