Journal of Data-centric Machine Learning Research (2024) Submitted 2/8; Revised 3/4; Published 5/11

Unveiling the Intertwined Relationship Between Essential Sparsity and Robustness in Large Pre-trained Models

Saebyeol Shin TOQUF930@G.SKKU.EDU
Department of Computer Science and Engineering
Sungkyunkwan University

Ajay Jaiswal AJAYJAISWAL@UTEXAS.EDU
Shiwei Liu SHIWEI.LIU@UTEXAS.EDU
Zhangyang Wang ATLASWANG@UTEXAS.EDU
Department of Electrical and Computer Engineering
The University of Texas at Austin

Reviewed on OpenReview: *https://openreview.net/forum?id=hN9vWadAYH*

Abstract

In the era of pre-trained LLMs, understanding their intrinsic sparse patterns becomes paramount, especially in the context of their scalability and efficiency. Recently, Jaiswal et al. (2023a) coined the concept of "essential sparsity" (ES) which states the existence of a sharp turning point in the sparsity-performance curve when large pre-trained models are pruned using simple magnitude-based criteria. Despite significant attention to investigating how pruning impacts the performance of pre-trained models, its impact on adversarial robustness and distribution shifts has been overlooked. In this work, we extend the concept of ES to robustness $\texttt{ES}_{\texttt{robust}}$, which illustrates the existence of a sharp turning point for robust performance. In comparison with clean performance, we found that sparsity tends to positively benefit the robust performance and $\texttt{ES}_{\texttt{robust}}$ is observed at slightly higher sparsity than ES. Our study presents a simple yet intriguing message that simple one-shot low-magnitude pruning is a powerful tool for identifying subnetworks that not only retain true performance but also robust performance on adversarial benchmarks. In addition, we found that carefully designed weight-importance criteria can further push the $\texttt{ES}_{\texttt{robust}}$ to non-trivial sparsity ratios (*e.g.* 50-55%). Moreover, we also extended our experiments across popular textual attacks (*e.g.,* deletion, character swap, etc.) for distribution shifts, and found our observations related to $\texttt{ES}_{\texttt{robust}}$ holds. All related codes will be open-sourced.

Keywords: Pruning, Adversarial Robustness, Distribution Shifts, Pre-trained Models

1 Introduction

Large-scale pre-trained language models have reshaped the paradigms of deep learning, setting new benchmarks across diverse applications from computer vision (Dosovitskiy et al., 2020; Han et al., 2022; Hugo et al., 2021; Parmar et al., 2018; Zheng et al., 2020) to natural language processing (Yang et al., 2019b; Liu et al., 2019; Talmor et al., 2018; Yang et al., 2019a; Wang et al., 2018; Ding et al., 2019; Chowdhery et al., 2022; Wei et al., 2022). Yet, as the parameter of these models exponentially increased, concerns regarding their efficiency, scalability, and real-world deployment have arisen, especially in environments where resources are constrained. To mitigate the high computational and memory footprints of

these models, *network pruning* which shrinks network sizes by removing specific weight from the model - essentially setting them to zero, has become one prominent research direction. Despite numerous existing algorithms (Frankle and Carbin, 2018; Chen et al., 2020; Jaiswal et al., 2021b; Yin et al., 2023a; You et al., 2019; Jaiswal et al., 2022; Lee et al., 2018; Yu et al., 2020), it is worth highlighting that these techniques have been notably effective for smaller networks, but their application to large language models remains challenging. The iterative processes involved in traditional pruning approaches, such as the Lottery Ticket Hypothesis (LTH) (Frankle et al., 2019), become increasingly infeasible as model sizes increase, due to computational overheads.

Recently, (Jaiswal et al., 2023a) proposed the concept of *essential sparsity* for large pre-trained models, which suggest that a significant proportion of the weights in them can be removed for free, although the proportion may vary depending on the complexity of the downstream task. This observation highlights the ease of removing parameters exploiting the emerged sparse patterns defined by low-magnitude weights, during pre-training at no cost. Interestingly, despite enormous attention towards achieving better efficiency without sacrificing performance, very limited attention has been given towards understanding the robustness inhibited by these compressed models. Some recent studies (Jin et al., 2020; Wang et al., 2019; Li et al., 2020) reveal that even dense large-scale language models are vulnerable to carefully crafted adversarial examples, which can fool the models to output arbitrarily wrong answers by perturbing input sentences in a human-imperceptible way. To this end, in this work, we investigate an **underexplored** direction: *How does compression induced by removing low-magnitude weights impact the robustness of dense models when evaluated on the unified adversarial benchmark?*

To this end, we **first** explore if pruning with low-magnitude weights preserve the robustness of large pre-trained models using exiting high-quality multi-task benchmarks for robustness evaluation of language models. Our work can be viewed as an extension of Essential Sparsity (Jaiswal et al., 2023a) for Robustness against the adverserial data. Based on our experimental observations, we define **essential sparsity for robustness** as the sharp dropping point, beyond which the robust performance of large pre-trained models drops significantly w.r.t. change in sparsity ratio. Our extensive experiments unveils several subtler and interesting findings:

- We found that simple one-shot low-magnitude pruning is a powerful tool for identifying subnetworks which not only retain true performance (estimated on clean benchmarks *e.g. GLUE*) but also robust performance on adversarial benchmarks.

- Similar to the observations in Jaiswal et al. (2023a), we found the existence of a sharp drop point for the robust performance, which we term as Essential Sparsity for Robustness ($\texttt{ES}_{\texttt{robust}}$). High Spearman's rank co-relation between true performance and robust performance unveils a two-in-one favourable quality of subnetworks identified by free-of-cost one-shot low-magnitude pruning.

- Our experiments across carefully selected distribution shifts (*e.g.* back-translation, deletion, character swap, etc.) illustrate that $\texttt{ES}_{\texttt{robust}}$ remains consistent across downstream tasks but the performance can vary depending on the strength of distribution

shift. This highlights the unique ability of subnetworks identified below $\texttt{ES}_{\texttt{robust}}$ to be resistant to various distribution shifts.

- Based on our extended experiments beyond one-shot low-magnitude pruning to recent other weight-importance based one-shot criteria like SparseGPT (Frantar and Alistarh, 2023) and Wanda (Sun et al., 2023), we found that a carefully selected importance criteria can push $\texttt{ES}_{\texttt{robust}}$ to significantly non-trivial high sparsity ratio (*50-55%* sparsity).

2 Experimental Setting

Datasets and Models: We engage with a diverse array of models for our analysis, including {$\texttt{BERT}_{\texttt{Base}}$(Devlin et al., 2018), $\texttt{Vicuna}_{\texttt{7B}}$(Chiang et al., 2023) $\texttt{Llama}_{\texttt{7B}}$(Touvron et al., 2023a), and $\texttt{Llama2}_{\texttt{7B}}$(Touvron et al., 2023b)}. For downstream NLP tasks, we select {SST-2, QNLI, and QQP} from the GLUE benchmark (Wang et al., 2018) as our clean dataset. To assess the vulnerability of language models under robust adversarial attacks in varied settings, we employ AdvGLUE++ (Wang et al., 2023), a dataset of challenging adversarial texts generated against open-source autoregressive models including Alpaca-7B (Taori et al., 2023), Vicuna-13B (Chiang et al., 2023), and StableVicuna-13B (Sta, 2023). We measure the model's accuracy on AdvGLUE++ data to determine its $\texttt{ES}_{\texttt{robust}}$, considering different adversarial text generation strategies. Additionally, we evaluate the ES on the corresponding benign data from GLUE, and measure the performance drop in comparison to the dense counterpart for various downstream tasks.

Pruning Method: We consider two types of sparsities: (1) *Unstructured Sparsity*: individual weights in the model are zeroed out independently, leading to irregular zero patterns (Han et al., 2015a); (2) *Structured N:M Sparsity*: a fine-grained sparsity pattern in which only N weights are non-zero for every continuous M weights (Zhou et al., 2021). We additionally include two more SoTA LLM pruning methods: SparseGPT (Frantar and Alistarh, 2023), and Wanda (Sun et al., 2023) to investigate how better weight importance criteria can benefit robustness of compressed models.

3 Existence of Essential Sparsity for Robustness

3.1 Revisiting Robustness for Pre-trained Language Models

The robustness of pre-trained language models has been a paramount concern, particularly when these systems are deployed in safety-critical applications such as autonomous vehicles (Roh et al., 2019; Yang et al., 2023), healthcare (Hu et al., 2023; Jaiswal et al., 2021a; Li et al., 2023), and cyber-security systems (Motlagh et al., 2024). LLMs have achieved state-of-the-art performance in a series of high-level natural language understanding (NLU) tasks, but the superior performance has only been observed in the benchmark test data that have the same distribution as the training set. Recent studies (Du et al., 2021; Niven and Kao, 2019; Utama et al., 2020) indicate that these LLMs are not robust and that the models do not remain predictive when the distribution of inputs changes. Specifically, these LLMs have low generalization performance when applied to out-of-distribution (OOD) test data and are also vulnerable to carefully crafted adversarial examples, which can fool the models to

output arbitrarily wrong answers by perturbing input sentences in a human-imperceptible way. To this end, various methods (Jiang et al., 2019; Liu et al., 2020; Wang et al., 2020a; Zhu et al., 2019) have explored imporving the adversarial robustness of language models. However, with increasing demand for model compression due to exponential growth of model size, it is surprising that robustness of compressed language models has been significantly overlooked. Our work delve into this underexplored direction and studies how language model compression using pruning impact the robustness of the model and its performance under distribution shifts.

3.2 Essential Sparsity for Robustness

The weights of a pruned language model can be depicted as $m \odot \theta$, where $m \in \{0,1\}^{|\theta|}$ is a binary mask with the same dimensionality as θ and $\odot$ is the elment-wise product. Let $\mathcal{E}^{\mathcal{T}}(f(x;\theta))$ denotes the robust performance of model $f(x;\theta)$ on the corresponding adverserial task $\mathcal{T}$. $\mathcal{P}_\rho(\cdot)$ is the sparsification algorithm which turns a portion ρ of "1" elements in the sparse mask m into "0"s. We extend the essential sparsity definition from Jaiswal et al. (2023a) to is a formal definition of essential sparsity for robustness ($\texttt{ES}_{\texttt{robust}}$) as following:

Essential Sparsity for Robustness. If $\mathcal{E}^{\mathcal{T}}(f(x; m \odot \theta)) \geq \mathcal{E}^{\mathcal{T}}(f(x;\theta)) - \epsilon$, and $\mathcal{E}^{\mathcal{T}}(f(x; \mathcal{P}_\rho(m) \odot \theta)) < \mathcal{E}^{\mathcal{T}}(f(x;\theta)) - \epsilon$ where the value of ρ and ϵ are small. Then, the according sparsity $1 - \frac{\|m\|_0}{|m|}$ is named as Essential Sparsity for Robustness of the model f on adversarial task $\mathcal{T}$.

As detailed above, the robust performance of model at $\texttt{ES}_{\texttt{robust}}$ usually has a turning point performance, which means further pruning even a small portion ρ of weights leads to at least ϵ performance drop on the adversarial dataset, compared to its dense counterpart $\mathcal{E}^{\mathcal{T}}(f(x,\theta))$. Note that in our work, we **do not** perform any adverserial training and all the performance is calculated using the clean pre-trained weights. In our case, ϵ is set as 1%.

Figure 1 presents the effect of pruning x% low-magnitude weights on the robust performance of four popular language models: BERT-base, Vicuna, Llama 1 & 2. Yellow bars indicate the performance of pruned models on the clean GLUE dataset while dashed lines indicate the robust performance on AdvGLUE++ (Wang et al., 2023), a dataset of challenging adversarial texts generated against open-source autoregressive models including Alpaca-7B (Taori et al., 2023), Vicuna-13B (Chiang et al., 2023), and StableVicuna-13B (Sta, 2023). Our observations include: ① we found that free-of-cost one-shot magnitude pruning to be a highly effective tool which can generate subnetworks that are highly robust to adversarial datasets, ② surprisingly, pruning language models seems to positively benefit improving robustness on our candidate adversarial datasets sometimes up to 30-40%, ③ across all our experiments, we found the existence of sharp turning point ($\texttt{ES}_{\texttt{robust}}$) which is downstream task-dependent and can vary depending on the task complexity, ④ we found a strong positive Spearman's rank co-relation between the clean performance and robust performance which indicate *robustness is free byproduct* while compressing language with simple models magnitude-based criterion, ⑤ it can be also observed that $\texttt{ES}_{\texttt{robust}}$ is *slightly higher* than $\texttt{ES}$.

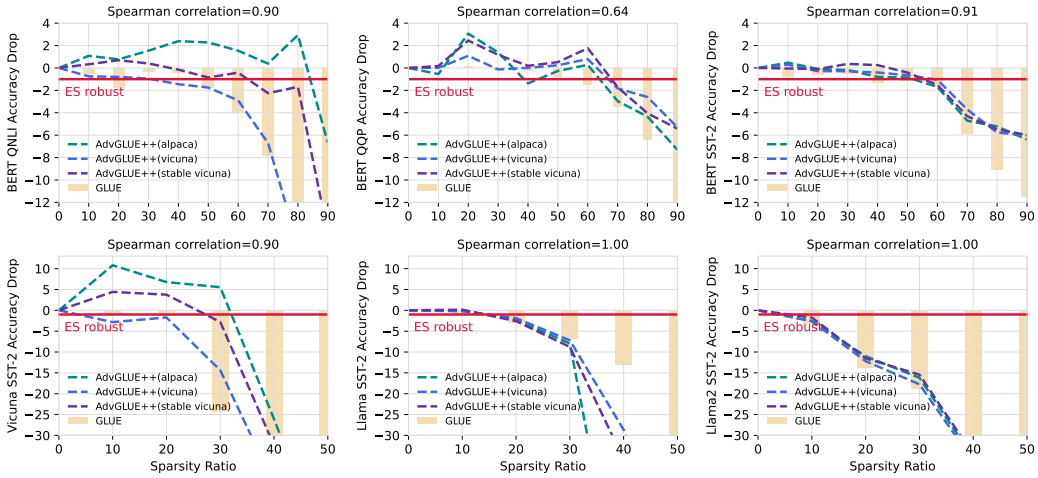

Figure 1: Performance drop estimated with respect to dense counterpart on downstream tasks using clean GLUE dataset and AdvGLUE++(Wang et al., 2023).

4 How does SoTA Weight-Importance Criterion impact $\texttt{ES}_{\texttt{robust}}$?

In modern LLMs, the presence of billions scale parameters restrict the adaptation of iterative prune-retrain-prune algorithms. To this end, SparseGPT (Frantar and Alistarh, 2023) and Wanda (Sun et al., 2023) are two new popular one-shot pruning algorithms with carefully crafted weight importance using calibration data. In this section, we analyze how a improved weight importance selection can benefit the robustness of the pruned subnetwork. Figure 3 presents the performance comparison of low-magnitude, sparsegpt, and wanda on our candidate adversarial datasets for N:M sparsity patterns and unstructured sparsity. Note that no fine-tuning has been performed for our results after pruning the model. Our observations can be summarized as: ① careful weight importance selection can significantly push the boundaries of $\texttt{ES}_{\texttt{robust}}$, ② we uniquely observe no performance degradation in robust performance up to a remarkable sparsity of 50%, ③ in comparison with wanda, we found sparsegpt to be more robustness friendly pruning method, ④ lastly, it is interesting to observe that Vicuna-7B can be pruned up to 40% with N:M sparsity with $\leq 5\%$ performance drop on some adversarial datasets.

5 Understanding $\texttt{ES}_{\texttt{robust}}$ under various Distribution Shifts

Recently, some work (Niven and Kao, 2019; Utama et al., 2020) identified that LLMs are highly sensitive for changes in the distribution of inputs, and their predictive behavior changes under distribution shifts. However, it is still underexplored how pruned models react to distribution shifts. In this section, we ask an important question: *How do different types of distribution shifts impact* $\texttt{ES}_{\texttt{robust}}$? To this end, we crafted 7 different distribution shifts (*e.g.* backtranslation, character swap, embedding-based transformation, wordnet synonyms replacement, etc.) from the GLUE benchmark. More details with examples can be found in Appendix A. We summarize our observations as follows: ① across all our candidate distribution shifts, we find the existence of essential sparsity for robustness and a common

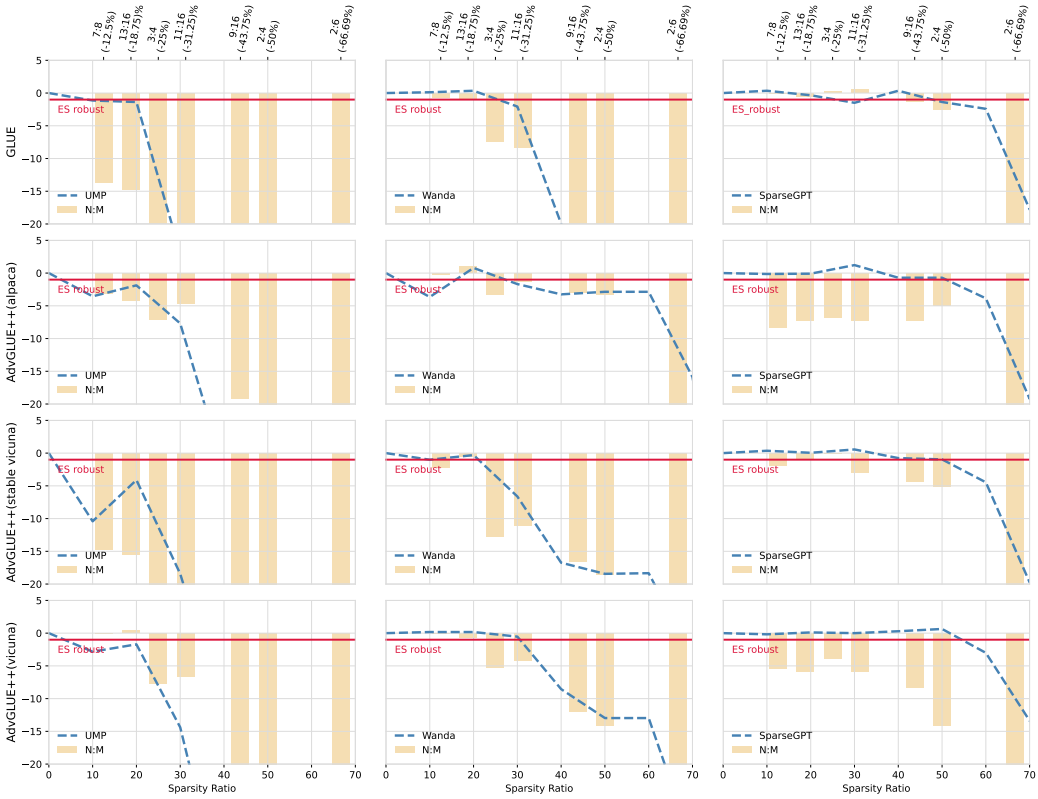

Figure 2: Performance drop estimated with respect to dense counterpart (Vicuna-7B) on downstream tasks using clean GLUE dataset and AdvGLUE++(Wang et al., 2023). Candidate models are compressed in one-shot using different weight importance criterions (low-magnitude, sparsegpt, wanda).

sharp turning point of the sparsity-performance curve, ② despite having a common $\texttt{ES}_{\texttt{robust}}$, performance of the compressed models can vary depending on the type of distribution shift, ③ both, sparsegpt and wanda pushes the $\texttt{ES}_{\texttt{robust}}$ from ∼20% sparsity to ∼30%, which send a strong single that simple one-shot pruning with careful weight importance estimation technique can be highly effective to retain the robustness of compressed models at higher sparsity ratios.

6 Impact of pruning calibration data on $\texttt{ES}_{\texttt{robust}}$

Many recent LLM pruning methods rely on calibration data to determine weight importance which form the basis of pruning. Among them, SparseGPT and Wanda are two popular methods which rely on `C4` (Raffel et al., 2019) instances. In this section, we investigated how the selection of the calibration dataset relates to the robust performance of the identified subnetwork. Figure 4 presents our experimental results for using calibration dataset from original `C4`, clean GLUE, and AdvGLUE++ for pruning the dense Vicuna-7B. Our results bring forth an interesting observation that selecting a calibration set from a similar distribution as the test set plays a vital role in retaining the robustness of the pruned model.

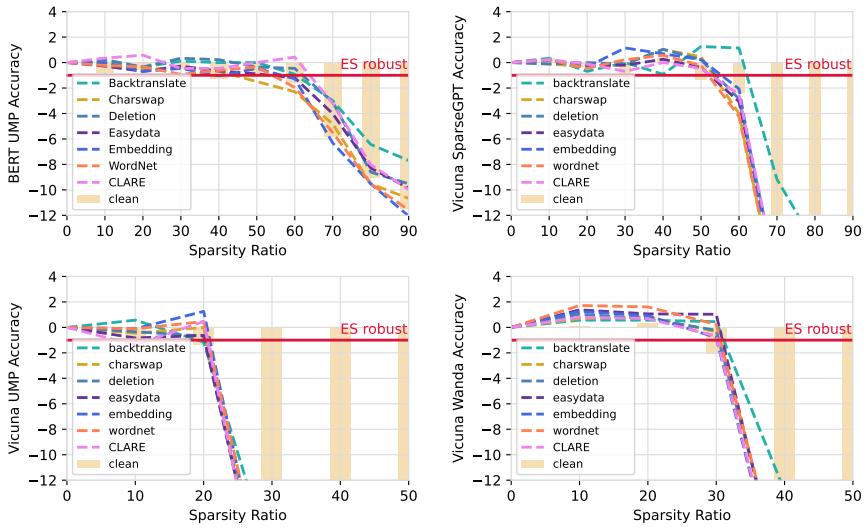

Figure 3: Performance drop estimated with respect to dense counterpart under different crafted distribution shifts from clean GLUE dataset. Candidate models are compressed in one shot using different weight importance criterions (low-magnitude, sparsegpt, wanda). Yellow bars indicate the performance on clean GLUE dataset.

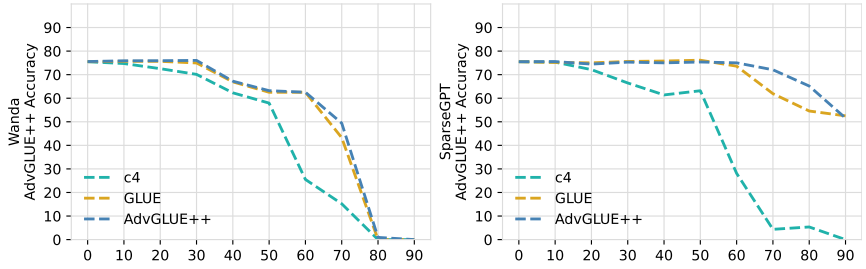

Figure 4: The performance of Vicuna-7B on AdvGLUE++ using different calibration data (c4, GLUE, AdvGLUE++) on Wanda and SparseGPT pruning method.

Contrary to the claim of insensitivity to calibration dataset in (Frantar and Alistarh, 2023; Sun et al., 2023) for clean performance, we observed both pruning algorithms have high sensitivity to selection of calibration dataset. Note that usage of AdvGLUE++ have slightly higher performance benefits than clean use for robust performance. Moreover, it can be observed that Wanda performance is comparatively less sensitive to calibration dataset in comparison with SparseGPT, despite they perform similarly using the `C4` dataset.

7 Related Work

7.1 Sparsity in Neural Networks.

Pruning LeCun et al. (1990); Han et al. (2015a) in deep neural networks, serves to simplify network architecture and enhance computational efficiency while aiming to preserve accuracy. This process can be categorized based on the sparsity patterns it introduces-unstructured sparsity Han et al. (2015a,b), which provides irregular distribution of non-zero

elements, and structured sparsity Liu et al. (2017); He et al. (2017); Zhou et al. (2016), where entire parameter groups, such as convolutional kernels or attention heads, are eliminated. While unstructured sparsity generally achieves superior performance due to its flexibility, structured sparsity tends to be more hardware-friendly. These sparsity patterns can be applied at various stages of the neural network's life cycle: post-training, during-training, and prior-training. Post-training sparsification, aimed at inference time efficiency, allows for significant pruning with minimal performance loss, often utilizing weight magnitude-based approaches as popularized by the Lottery Ticket Hypothesis Frankle and Carbin (2018). During-training sparsification Finnoff et al. (1993); Luo and Wu (2020); Savarese et al. (2020); Sehwag et al. (2020), on the other hand, aims for computational savings during the model training process itself, gradually introducing sparsity and potentially re-activating pruned connections later in training. Prior-training sparsity Lee et al. (2018); Tanaka et al. (2020); De Jorge et al. (2020); Wang et al. (2020b) involves identifying crucial sparse connectivities at the network's initialization.

7.2 Sparsity in LLM.

Pre-trained Transformers have solidified their status as the predominant choice across a wide array of natural language processing (NLP) applications (Yang et al., 2019b; Liu et al., 2019; Talmor et al., 2018; Chowdhery et al., 2022; Wei et al., 2022). Nonetheless, the computational cost of training these models is substantial, often requiring thousands of GPUs for extended periods (Brown et al., 2020). To mitigate these resource demands, extensive research efforts (Kurtic et al., 2022; Liu et al., 2023; Yin et al., 2023b,c; Lagunas et al., 2021; Jaiswal et al., 2023b) have been undertaken. In the realm of Large Language Models (LLMs), traditional pruning has faced challenges due to the necessity of re-training rounds to restore performance. However, recent advancements in LLM-specific pruning algorithms, such as SparseGPT (Frantar and Alistarh, 2023) and Wanda (Sun et al., 2023), have shown substantial progress. Moreover, a recent study (Jaiswal et al., 2023a) reveals the advantageous effects of essential sparsity induced during pre-training, demonstrating how we can leverage it to efficiently prune large pre-trained models without incurring additional computational costs. In this work, we aim to investigate whether essential sparsity correlates with adversarial robustness. The relationship remains unclear, particularly regarding which sparsity ratios are insensitive to significant drops in robustness. Therefore, understanding essential sparsity from the perspective of robustness is crucial.

8 Conclusion

In this comprehensive study, we have explored the relationship between $\texttt{ES}_{\texttt{robust}}$ and $\texttt{ES}$ across a spectrum of model sizes, ranging from BERT to Vicuna. Our investigation delved into the effects of multiple one-shot pruning methods with different weight importance criteria, including N:M sparsity patterns. We found that carefully designed weight-importance criteria can further push the $\texttt{ES}_{\texttt{robust}}$ to non-trivial sparsity ratios (*e.g.* 40-50%). Moreover, we also extended our experiments across popular textual attacks (*e.g.,* deletion, character swap, etc.) for distribution shifts, and found our observations related to $\texttt{ES}_{\texttt{robust}}$ holds. Our future work includes extending our study beyond robustness to other settings like fairness, interpretability, and bias.

References

Stablevicuna: An rlhf fine-tune of vicuna-13b v0 available at https://github.com/stabilityai/stablevicuna, 4 2023. url https://stability.ai/blog/stablevicuna-open-source-rlhf-chatbot. doi:10.57967/hf/0588, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. The lottery ticket hypothesis for pre-trained bert networks. *Advances in neural information processing systems*, 33:15834–15846, 2020.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2023.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Pau De Jorge, Amartya Sanyal, Harkirat S Behl, Philip HS Torr, Gregory Rogez, and Puneet K Dokania. Progressive skeletonization: Trimming more fat from a network at initialization. *arXiv preprint arXiv:2006.09081*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Ming Ding, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. Cognitive graph for multi-hop reading comprehension at scale. *arXiv preprint arXiv:1905.05460*, 2019.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Mengnan Du, Varun Manjunatha, Rajiv Jain, Ruchi Deshpande, Franck Dernoncourt, Jiuxiang Gu, Tong Sun, and Xia Hu. Towards interpreting and mitigating shortcut learning behavior of nlu models, 2021.

William Finnoff, Ferdinand Hergert, and Hans Georg Zimmermann. Improving model selection by nonconvergent methods. *Neural Networks*, 6(6):771–783, 1993.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel M Roy, and Michael Carbin. Stabilizing the lottery ticket hypothesis. *arXiv preprint arXiv:1903.01611*, 2019.

Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. 2023.

Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 45(1):87–110, 2022.

Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015a.

Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015b.

Yihui He, Xiangyu Zhang, and Jian Sun. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1389–1397, 2017.

Mingzhe Hu, Shaoyan Pan, Yuheng Li, and Xiaofeng Yang. Advancing medical imaging with language models: A journey from n-grams to chatgpt, 2023.

Touvron Hugo, Matthieu Cord, Douze Matthijs, Massa Francisco, Sablayrolles Alexandre, and Jegou Herve. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021.

Ajay Jaiswal, Liyan Tang, Meheli Ghosh, Justin Rousseau, Yifan Peng, and Ying Ding. Radbert-cl: Factually-aware contrastive learning for radiology report classification, 2021a.

Ajay Jaiswal, Shiwei Liu, Tianlong Chen, and Zhangyang Wang. The emergence of essential sparsity in large pre-trained models: The weights that matter. *arXiv preprint arXiv:2306.03805*, 2023a.

Ajay Kumar Jaiswal, Haoyu Ma, Tianlong Chen, Ying Ding, and Zhangyang Wang. Spending your winning lottery better after drawing it. *arXiv preprint arXiv:2101.03255*, 2021b.

Ajay Kumar Jaiswal, Haoyu Ma, Tianlong Chen, Ying Ding, and Zhangyang Wang. Training your sparse neural network better with any mask. In *International Conference on Machine Learning*, pages 9833–9844. PMLR, 2022.

Ajay Kumar Jaiswal, Shiwei Liu, Tianlong Chen, Ying Ding, and Zhangyang Wang. Instant soup: Cheap pruning ensembles in a single pass can draw lottery tickets from large models. In *International Conference on Machine Learning*, pages 14691–14701. PMLR, 2023b.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. Smart: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. 2019. doi: 10.18653/v1/2020.acl-main.197.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025, 2020.

Eldar Kurtic, Daniel Campos, Tuan Nguyen, Elias Frantar, Mark Kurtz, Benjamin Fineran, Michael Goin, and Dan Alistarh. The optimal bert surgeon: Scalable and accurate second-order pruning for large language models. *arXiv preprint arXiv:2203.07259*, 2022.

François Lagunas, Ella Charlaix, Victor Sanh, and Alexander M Rush. Block pruning for faster transformers. *arXiv preprint arXiv:2109.04838*, 2021.

Yann LeCun, B Boser, John S Denker, Donnie Henderson, RE Howard, Wayne E Hubbard, LD Jackel, and DS Touretzky. Advances in neural information processing systems. *San Francisco, CA, USA: Morgan Kaufmann Publishers Inc*, pages 396–404, 1990.

Namhoon Lee, Thalaiyasingam Ajanthan, and Philip HS Torr. Snip: Single-shot network pruning based on connection sensitivity. *arXiv preprint arXiv:1810.02340*, 2018.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. Bert-attack: Adversarial attack against bert using bert, 2020.

Tianhao Li, Sandesh Shetty, Advaith Kamath, Ajay Jaiswal, Xianqian Jiang, Ying Ding, and Yejin Kim. Cancergpt: Few-shot drug pair synergy prediction using large pre-trained language models, 2023.

Shiwei Liu, Tianlong Chen, Zhenyu Zhang, Xuxi Chen, Tianjin Huang, Ajay Jaiswal, and Zhangyang Wang. Sparsity may cry: Let us fail (current) sparse neural networks together! *arXiv preprint arXiv:2303.02141*, 2023.

Xiaodong Liu, Hao Cheng, Pengcheng He, Weizhu Chen, Yu Wang, Hoifung Poon, and Jianfeng Gao. Adversarial training for large neural language models. *arXiv preprint arXiv:2004.08994*, 2020.

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

Zhuang Liu, Jianguo Li, Zhiqiang Shen, Gao Huang, Shoumeng Yan, and Changshui Zhang. Learning efficient convolutional networks through network slimming. In *Proceedings of the IEEE international conference on computer vision*, pages 2736–2744, 2017.

Jian-Hao Luo and Jianxin Wu. Autopruner: An end-to-end trainable filter pruning method for efficient deep model inference. *Pattern Recognition*, 107:107461, 2020.

Farzad Nourmohammadzadeh Motlagh, Mehrdad Hajizadeh, Mehryar Majd, Pejman Najafi, Feng Cheng, and Christoph Meinel. Large language models in cybersecurity: State-of-the-art, 2024.

Timothy Niven and Hung-Yu Kao. Probing neural network comprehension of natural language arguments, 2019.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Łukasz Kaiser, and Noam Shazeer. Alexander ku and dustin tran. image transformer. *arXiv preprint arXiv: 1802.05751*, 2018.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv e-prints*, 2019.

Junha Roh, Chris Paxton, Andrzej Pronobis, Ali Farhadi, and Dieter Fox. Conditional driving from natural language instructions, 2019.

Pedro Savarese, Hugo Silva, and Michael Maire. Winning the lottery with continuous sparsification. *Advances in neural information processing systems*, 33:11380–11390, 2020.

Vikash Sehwag, Shiqi Wang, Prateek Mittal, and Suman Jana. Hydra: Pruning adversarially robust neural networks. *Advances in Neural Information Processing Systems*, 33:19655–19666, 2020.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*, 2023.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*, 2018.

Hidenori Tanaka, Daniel Kunin, Daniel L Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. *Advances in neural information processing systems*, 33:6377–6389, 2020.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Stanford alpaca: An instruction-following llama model, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. Towards debiasing nlu models from unknown biases, 2020.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Boxin Wang, Hengzhi Pei, Boyuan Pan, Qian Chen, Shuohang Wang, and Bo Li. T3: Tree-autoencoder constrained adversarial text generation for targeted attack, 2019.

Boxin Wang, Shuohang Wang, Yu Cheng, Zhe Gan, Ruoxi Jia, Bo Li, and Jingjing Liu. Infobert: Improving robustness of language models from an information theoretic perspective. *arXiv preprint arXiv:2010.02329*, 2020a.

Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. *arXiv preprint arXiv:2306.11698*, 2023.

Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. *arXiv preprint arXiv:2002.07376*, 2020b.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. End-to-end open-domain question answering with bertserini. *arXiv preprint arXiv:1902.01718*, 2019a.

Zhenjie Yang, Xiaosong Jia, Hongyang Li, and Junchi Yan. Llm4drive: A survey of large language models for autonomous driving, 2023.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019b.

Lu Yin, Shiwei Liu, Meng Fang, Tianjin Huang, Vlado Menkovski, and Mykola Pechenizkiy. Lottery pools: Winning more by interpolating tickets without increasing training or inference cost. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 10945–10953, 2023a.

Lu Yin, Shiwei Liu, Ajay Jaiswal, Souvik Kundu, and Zhangyang Wang. Junk dna hypothesis: A task-centric angle of llm pre-trained weights through sparsity. *arXiv preprint arXiv:2310.02277*, 2023b.

Lu Yin, You Wu, Zhenyu Zhang, Cheng-Yu Hsieh, Yaqing Wang, Yiling Jia, Mykola Pechenizkiy, Yi Liang, Zhangyang Wang, and Shiwei Liu. Outlier weighed layerwise sparsity (owl): A missing secret sauce for pruning llms to high sparsity. *arXiv preprint arXiv:2310.05175*, 2023c.

Haoran You, Chaojian Li, Pengfei Xu, Yonggan Fu, Yue Wang, Xiaohan Chen, Richard G Baraniuk, Zhangyang Wang, and Yingyan Lin. Drawing early-bird tickets: Towards more efficient training of deep networks. *arXiv preprint arXiv:1909.11957*, 2019.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020.

Minghang Zheng, Peng Gao, Renrui Zhang, Kunchang Li, Xiaogang Wang, Hongsheng Li, and Hao Dong. End-to-end object detection with adaptive clustering transformer. *arXiv preprint arXiv:2011.09315*, 2020.

Aojun Zhou, Yukun Ma, Junnan Zhu, Jianbo Liu, Zhijie Zhang, Kun Yuan, Wenxiu Sun, and Hongsheng Li. Learning n: m fine-grained structured sparse neural networks from scratch. *arXiv preprint arXiv:2102.04010*, 2021.

Hao Zhou, Jose M Alvarez, and Fatih Porikli. Less is more: Towards compact cnns. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 662–677. Springer, 2016.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. Freelb: Enhanced adversarial training for natural language understanding. *arXiv preprint arXiv:1909.11764*, 2019.

Appendix A. Example of Distribution Shifts

Original sentence
"What I cannot create, I do not understand."

Backtranslate
"What you can't create, you don't understand."
Sentence level augmentation that uses MarianMTModel to back-translate. Romance language (French, Italian, Portuguese, Spanish etc) to English

Charswap
"What I cLnnot create, I do not understand."
Augments words by swapping characters out for other characters.

EasyData
"What I cannot create, I do understand.'
"create I cannot What"
"I do not understand."
"What I cannot create, I ane do not understand."
"What I cannot create, I do not see."
An implementation of Easy Data Augmentation, which combines:

- *WordNet synonym replacement* (Randomly replace words with their synonyms.)
- *Word deletion* (Randomly remove words from the sentence.)
- *Word order swaps* (Randomly swap the position of words in the sentence.)
- *Random synonym insertion* (Insert a random synonym of a random word at a random location.)

Embedding
"Whereof I cannot create, I do not understand."
Augments text by transforming words with their embeddings.

WordNet
"What I cannot create, I do not empathise."
Augments text by replacing with synonyms from the WordNet thesaurus.

CLARE
"What I cannot create, I purposely do not understand."
CLARE builds on a pre-trained masked language model and modifies the inputs in a contextaware manner. Three contextualized perturbations, Replace, Insert and Merge, allowing for generating outputs of varied lengths.