Quantization vs Pruning: Insights from the Strong Lottery Ticket Hypothesis

Anonymous Author(s)
Affiliation
Address
email

Abstract

Quantization is an essential technique for making neural networks more efficient, yet our theoretical understanding of it remains limited. Previous works demonstrated that extremely low-precision networks, such as binary networks, can be constructed by pruning large, randomly-initialized networks, and showed that the ratio between the size of the original and the pruned networks is at most polylogarithmic.

The specific pruning method they employed inspired a line of theoretical work known as the Strong Lottery Ticket Hypothesis (SLTH), which leverages insights from the Random Subset Sum Problem. However, these results primarily address the continuous setting and cannot be applied to extend SLTH results to the quantized setting.

In this work, we build on foundational results by Borgs et al. on the Number Partitioning Problem to derive new theoretical results for the Random Subset Sum Problem in a quantized setting. Using these results, we then extend the SLTH framework to finite-precision networks. While prior work on SLTH showed that pruning allows approximation of a certain class of neural networks, we demonstrate that, in the quantized setting, the analogous class of target discrete neural networks can be represented exactly, and we prove optimal bounds on the necessary over-parameterization of the initial network as a function of the precision of the target network.

1 Introduction

Deep neural networks (DNNs) have become ubiquitous in modern machine-learning systems, yet their ever-growing size quickly collides with the energy, memory, and latency constraints of real-world hardware. **Quantization**—representing weights with a small number of bits—is arguably the most hardware-friendly compression technique, and recent empirical work shows that aggressive quantization can preserve accuracy even down to the few bits regime. Unfortunately, our theoretical understanding of why and when such extreme precision reduction is possible still lags far behind practice. An interesting step in this direction was the *Multi-prize Lottery Ticket Hypothesis* (MPLTH) put forward by Diffenderfer and Kailkhura [2021]. They *empirically* demonstrated that a sufficiently large, randomly initialized network contains sparse *binary* subnetworks that match the performance of a target network with real-valued weights. They also provided theoretical guarantees regarding the existence of such highly quantized networks, showing that, with respect to the target network, the initial random network need only be larger by a **polynomial factor**. Sreenivasan et al. [2022] subsequently improved this bound, by showing that a polylogarithmic factor is sufficient (See Section 2). These works fall within the research topic known as the *Strong Lottery Ticket Hypothesis* (SLTH), which states that sufficiently-large randomly initialized neural networks contain subnetworks, called

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

lottery tickets, that perform well on a given task, without requiring weight adjustments. The main theoretical question, therefore, is: how large should the initial network be to ensure it contains a lottery ticket capable of approximating a given family of target neural networks? Research on the SLTH, however, has mainly focused on investigating pruning in the continuous-weight (i.e. not quantized) setting, drawing on results for the Random Subset Sum Problem (RSSP) [Lueker, 1998] to show that over-parameterized networks can be pruned to *approximate* any target network without further training [Orseau et al., 2020, Pensia et al., 2020, Burkholz, 2022a] (for additional context on this body of literature, we kindly refer the reader to the Related Work in Section 2). However, the analytic RSSP results used for SLTH rely heavily on real-valued weights and therefore do not extend to the finite-precision regime considered in the MPLTH. This gap left open a fundamental question:

What is the over-parameterization needed to obtain *quantized* strong lottery tickets?

Our contributions. We address the aforementioned gap by revisiting the classic *Number Partitioning Problem* (NPP), which is closely related to the RSSP. Building on the seminal results of Borgs et al. [2001] concerning the *phase transition* of NPP, we derive new, sharp bounds for the **discrete RSSP**. These bounds are precise enough to adapt the SLTH proof strategy to the finite precision setting and, in doing so, establish optimal bounds for the MPLTH. Crucially, our results account for arbitrary quantization in both the initial and target networks, and demonstrate that the lottery ticket can represent the target network *exactly*. In contrast, prior work limited the initial network to binary weights and assumed continuous weights for the larger (target) network [Diffenderfer and Kailkhura, 2021, Sreenivasan et al., 2022], requiring a cubic overparameterization relative to the lower bound and additional dependencies on network parameters absent in our bound. Concretely, let δ_t denote the precision (i.e., quantization level) of a target network N_t, δ_{in} the precision of a randomly initialized larger network N_{in}, and δ any parameter satisfying $\delta_t \geq \delta^2 \geq \delta_{\text{in}}^2$. Denote by d and ℓ the width and depth of the target network, respectively. Our results can be summarized by the following simplified, informal theorem (refer to the formal statements for full generality).

Theorem (Informal version of Theorems 1, 2, and 3)**.** *With high probability*[1]*, a depth-2ℓ network N_{in} of width $\mathcal{O}\big(d\log(1/\delta)\big)$ can be quantized to precision δ and pruned to become functionally equivalent to any δ_t-quantized target network N_t with layers of width at most d (Theorem 1). This result is optimal, as no two-layer network of precision δ with fewer than $\Omega\big(d\log(1/\delta)\big)$ parameters can be pruned to represent δ-quantized neural networks of width d (Theorem 3). Furthermore, the depth of N_{in} can be reduced to $(\ell+1)$ at the cost of an additional $\log(1/\delta)$ factor in its width (Theorem 2).*

These are the first theoretical results that (i) characterize the precise interplay between weight precision and over-parameterization, and (ii) certify that pruning can yield *exact*, not just approximate, quantized subnetworks. Besides contributing to the theory of network compression, our analysis showcases the versatility of classical combinatorial insights—such as the theory of NPP—in deep-learning theory.

Paper organization. In Section 2, we review prior work on SLTH and quantization. In Section 3, we prove a new quantized version of the RSSP, after first recalling classical results on RNPP in subsection 3.1. Our new theorems on the quantized SLTH are proved in Sections 4, after recalling necessary notation and definitions in subsection 4.1. Finally, in Section 5, we discuss the conclusion of our work and future directions.

2 Related Work

Strong Lottery Ticket Hypothesis. In 2018, Frankle and Carbin [2019] proposed the Lottery Ticket Hypothesis, which states that every dense network contains a sparse subnetwork that can be trained from scratch, and performs equally well as the dense network. Rather surprisingly, Zhou et al. [2019], Ramanujan et al. [2020] and Wang et al. [2019] empirically showed that it is possible to efficiently find subnetworks within large randomly initialized networks that perform well on a given task, without changing the initial weights. This motivated the Strong Lottery Ticket Hypothesis (SLTH), which states that sufficiently overparameterized randomly initialized neural networks contain sparse subnetworks that will perform as well as a small trained network on a given dataset, without

[1]As customary in the literature on randomized algorithms, with the expression *with high probability* we refer to a probability of failure which scales as the inverse of a polynomial in the parameter of interest (the number of precision bits $\log 1/\delta$) in our case).

any training. Many formal results rigorously proved the SLTH in various settings, the first one being Malach et al. [2020], where they showed that a feed-forward dense target network of width ℓ and depth d can be approximated by pruning a random network of depth 2ℓ and width $\mathcal{O}(d^5\ell^2)$. Orseau et al. [2020], Pensia et al. [2020] improved this bound by proving that width $\mathcal{O}(d\log(d\ell))$ is sufficient. Another construction was provided by Burkholz [2022a], where they showed that a network of width $\ell+1$ is enough to approximate a network of width ℓ, with a certain compromise on the width. Other works extended the SLTH to other famous architectures, such as convolutional Burkholz [2022b] and equivariant networks Ferbach et al. [2022]. Next, we provide an informal version that qualitatively summarizes this kind of results.

Theorem (Informal qualitative template of SLTH results)**.** *With high probability, a random artificial neural network N_R with m parameters can be pruned so that the resulting subnetwork approximates, up to an error ϵ, any target artificial neural network N_T with $\mathcal{O}(m/\log_2(1/\epsilon))$ parameters. The logarithmic dependency on ϵ is optimal.*

Quantization. Neural network quantization refers to the process of reducing the precision of the weights within a neural network. Empirical studies have demonstrated that trained neural networks can often be significantly quantized without incurring substantial loss in performance Han et al. [2015]. In particular, Diffenderfer and Kailkhura [2021] provided both empirical and theoretical support for a quantized variant of the SLTH, introducing an algorithm capable of training binary networks effectively. With regard to theoretical guarantees, they proved that a neural network with width d and depth ℓ can be approximated to within an error ϵ by a binary target network of width $\mathcal{O}(\ell d^{3/2}/\epsilon + \ell d\log(\ell d/\epsilon))$. Subsequently, Sreenivasan et al. [2022] presented an exponential improvement over this result, demonstrating that a binary network with depth $\Theta(\ell\log(d\ell/\epsilon))$ and width $\Theta(d\log^2(d\ell/\epsilon))$ suffices to approximate any given network of width d and depth ℓ. We remark that both of these results assume that the initial network weights are binary, whereas the target network weights are continuous. The success of techniques to construct heavily quantized networks can be related to theoretical work that show that heavily quantized networks still retain good universal approximation properties [Hwang et al., 2024]. In practice, not all parts of a network need to be quantized equally aggressively. Mixed-precision quantization allocates different bit-widths to different layers or parameters to balance accuracy and efficiency [Carilli, 2020, Younes Belkada, 2022].

Subset Sum Problem (SSP). Given a target value z and a multiset Ω of n integers from the set $\{-M, -M+1, \ldots, M-1, M\}$, the SSP consists in finding a subset $S \subseteq \Omega$ such that the sum of its elements equal z. In Computational Complexity Theory, SSP is one of the most famous NP-complete problems Garey and Johnson [1979]. Its random version, Random SSP (RSSP), has been investigated since the 80s in the context of combinatorial optimization [Lueker, 1982, 1998], and recently received renowned attention in the machine learning community because of its connection to the SLTH [Pensia et al., 2020].

Number Partitioning Problem (NPP). NPP is the problem of partitioning a multiset Ω of n integers from the set $[M] := \{1, 2, \ldots, M\}$ into two subsets such that the difference of their respective sums equals a target value z (typically, the literature has focused on minimizing $|z|$, i.e. trying to approximate the value closest to zero). Analogously to the aforementioned SSP, NPP is one of the most important NP-complete problems [Garey and Johnson, 1979, Hayes, 2002]. Its random version, in which the n elements are sampled uniformly at random from $[M]$, has also received considerable attention in Statistical Physics, where it has been shown to exhibit a phase transition Mézard and Montanari [2009]. Concretely, Mertens [1998] heuristically showed the following result, which was later put on rigorous grounds by Borgs et al. [2001]: defining $\kappa := \frac{\log_2 M}{n}$, if $\kappa < 1$ then $\mathcal{O}(2^n)$ number of solutions exist, whereas if $\kappa > 1$ the number of solutions sharply drops to zero.

3 Quantized Random Subset Sum Problem

3.1 Random Number Partitioning Problem

In this section, we recall seminal results by Borgs et al. [2001] which we leverage in Subsection 3.2 to obtain new results on the quantized RSSP.

Definition 1 (RNNP)**.** *Let $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ be a set of integers sampled uniformly from the set $\{1, 2, \ldots, M\}$. The Random Number Partitioning Problem is defined as the problem of finding a partitioning set $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ with $\sigma_j \in \{-1, 1\}$ such that $|\boldsymbol{\sigma} \cdot \boldsymbol{x}| = z$ for some given integer z (called target).*

Note that usually, in RNPP the difference between the sum of two parts is minimized, but we consider RNPP with a target, i.e., the difference between the sum of two parts must be equal to a given number z, the target. Given an instance of Random Number Partitioning Problem $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ with a set of size n and a target z, $Z_{n,z}$ denotes the number of exact solutions to the RNPP, i.e.,

$$Z_{n,z} = \sum_{\boldsymbol{\sigma}} \mathbb{I}(|\boldsymbol{\sigma} \cdot \boldsymbol{X}| = z).$$

To prove the existence of phase transition, Borgs et al. [2001] estimated the moments of $Z_{n,z}$. The relevant result is stated as Theorem 4 (Appendix A). Using these moment estimates of $Z_{n,z}$, one can write an upper and a lower bound on the probabilities of existence of solutions to a RNPP. We do so in Lemma 1.

Lemma 1. *Given a Random Number Partitioning Problem, the probability $\mathbb{P}(Z_{n,z} > 0)$ is bounded above and below as*

$$\mathbb{P}(Z_{n,z} > 0) \leq \begin{cases} \frac{\rho_n}{2} \left(\exp\left(-\frac{z^2}{2nM^2 c_M} \right) + \mathcal{O}\left(\frac{1}{n}\right) \right) & if \;\; z = 0 \\ \rho_n \left(\exp\left(-\frac{z^2}{2nM^2 c_M} \right) + \mathcal{O}\left(\frac{1}{n}\right) \right) & if \;\; z \neq 0 \end{cases}$$

$$\mathbb{P}(Z_{n,z} > 0) \geq \frac{1}{2\left(1 + \exp\left(\frac{z^2}{nM^2 c_M}\right)\left(\mathcal{O}\left(\frac{1}{n\rho_n}\right) + \mathcal{O}\left(\frac{1}{n}\right)\right) + \frac{1}{\rho_n}\right)}$$

where ρ_n is defined as $\rho_n = 2^{n+1}\gamma_n$.

Proof Sketch. We use Markov's Inequality (Theorem 5 in Appendix C) and Cauchy-Schwartz inequality (Theorem 6 in Appendix C) to get bounds on the probabilities from the moment estimates (Theorem 4). See Appendix A for details. □

The existence of phase transition (Section 2) is a consequence of Lemma 1 but for the purposes of this paper, we only require Lemma 1.

3.2 Quantized Random Subset Sum Problem

The Random Subset Sum Problem (RSSP) is the problem of finding a subset of a given set such that the sum of this subset equals a given target t. RSSP is a crucial tool in proving results on SLTH Pensia et al. [2020] Burkholz [2022a]. RSSP and RNPP are closely related, and hence we can use the results on RNPP in this section to make statements about RSSP. We shall then use these results on RSSP to prove results on SLTH and quantization.

Definition 2 (RSSP)**.** *Let $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ be a set of integers sampled uniformly from the set $\{-M, \ldots, 1, 2, 3, \ldots, M\}$. The RSSP is defined as the problem of finding an index set $S \subset [n]$ such that $\sum_{i \in S} X_i = t$ for a given integer t, called the target.*

Lemma 2. *An SSP with given set $\mathbf{X} = (X_1, X_2, \ldots, , X_n)$ and target t can be solved iff the NPP can be solved on the given set $\mathbf{X}$ and target $\Lambda - 2t$ (or $2t - \Lambda$), where $\Lambda = \sum_{i=1}^{n} X_i$.*

Lemma 2 is proved in Appendix A. Using the equivalence of RNPP and RSSP, the following results on RSSP follows from Lemma 1.

Lemma 3. *Consider a RSSP on the set $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ where X_i's are sampled uniformly from $\{-M, \ldots, -1, 1, \ldots, M\}$ with a target $t = \mathcal{O}(M)$. Let $Y_{n,t}$ be the number of possible solutions to the RSSP problem. Then*

$$\mathbb{P}(Y_{n,t} > 0) \leq \begin{cases} \rho_n \left(\exp\left(-\frac{z^2}{2nM^2 c_M} \right) + \mathcal{O}\left(\frac{1}{n}\right) \right) & if \;\; z = 0 \\ 2\rho_n \left(\exp\left(-\frac{z^2}{2nM^2 c_M} \right) + \mathcal{O}\left(\frac{1}{n}\right) \right) & if \;\; z \neq 0, \end{cases}$$

$$\mathbb{P}(Y_{n,t} > 0) \geq \frac{1}{\left(1 + \exp\left(\frac{z^2}{nM^2 c_M}\right)\right)\left(\mathcal{O}\left(\frac{1}{n\rho_n}\right) + \mathcal{O}\left(\frac{1}{n}\right)\right) + \frac{1}{\rho_n}},$$

where

$$z = \Lambda - 2t, \qquad \Lambda = \sum_{i=1}^{n} X_i,$$

$$\gamma_n = \frac{1}{M\sqrt{2\pi n c_M}}, \qquad c_M = \mathbb{E}\left(\frac{X^2}{M^2}\right) = \frac{1}{3} + \frac{1}{2M} + \frac{1}{6M^2}.$$

Proof Sketch. We first convert the given RSSP to a RNPP through the transformation in Lemma 2. The result then follows from Lemma 1. See Appendix A for details. □

The next lemma shows under what condition an RSSP can be solved with high probability.

Lemma 4. *Let $M = M(n)$ be an arbitrary function of n. Consider as RSSP on the set $\mathbf{X} = (X_1, X_2, \dots, X_n)$ sampled uniformly from $\{-M, \dots, -1, 1, \dots, M\}$ with a target $t = \mathcal{O}(M)$. If*

$$\kappa_n = \lim_{n\to\infty} \frac{\log_2 M}{n} < 1,$$

then we have

$$\mathbb{P}(Y_{n,t} > 0) = 1 - \mathcal{O}\left(\frac{1}{n^{\frac{1}{7}}}\right).$$

Proof Sketch. It can be shown using Hoeffding's inequality (Theorem 7 in Appendix C), that with high probability the sum of all elements satisfies $\Lambda < \sqrt{\frac{2}{7}} M\sqrt{n \log n}$. Hence, the probability of solving a RSSP from Lemma 3 can be analyzed under the assumption of $\kappa_n < 1$. See Appendix A for details. □

4 SLTH and Weight Quantization

4.1 Notation and Setup

In this Subsection, we define some notation before stating our results. Scalars are denoted by lowercase letters such as w, y, etc. Vectors are represented by bold lowercase letters, e.g., $\mathbf{v}$, and the i^{th} component of a vector $\mathbf{v}$ is denoted by v_i. Matrices are denoted by bold uppercase letters such as $\mathbf{M}$. If a matrix $\mathbf{W}$ has dimensions $d_1 \times d_2$, we write $\mathbf{W} \in \mathbb{R}^{d_1 \times d_2}$. We define the finite set $S_\delta := \{-1, -1+\delta, -1+2\delta, \dots, 1\}$, where $\delta = 2^{-k}$ for some $k \in \mathbb{N}$. A real number b is said to have precision δ if $b \in S_\delta$. We denote the d -fold Cartesian product of S_δ by S_δ^d; that is,

$$S_\delta^d := \underbrace{S_\delta \times \cdots \times S_\delta}_{d \text{ times}}.$$

For $w \in S_\delta$ with $\delta = 2^{-k}$ and $\gamma = 2^{-m}$ such that $k > m$, we define the quantization operator $[\cdot]_\gamma$ by

$$[w]_\gamma := \frac{\lfloor w 2^m \rfloor}{2^m}.$$

This operation reduces the precision of w to γ. For a vector $\mathbf{v}$, the notation $\mathbf{w} = [\mathbf{v}]_\gamma$ means $w_i = [v_i]_\gamma$ for all components i. We use C, C_i for $i \in \mathbb{N}$ to denote positive absolute constants.

Definition 3. *An ℓ-layer neural network is a function $f : \mathbb{R}^{d_0} \to \mathbb{R}^{d_\ell}$ defined as*

$$f(\mathbf{x}) := \mathbf{W}_\ell \sigma(\mathbf{W}_{\ell-1} \cdots \sigma(\mathbf{W}_1 \mathbf{x})), \tag{1}$$

where $\mathbf{W}_i \in \mathbb{R}^{d_i \times d_{i-1}}$ for $i = 1, \dots, \ell$, $\mathbf{x} \in \mathbb{R}^{d_0}$, and $\sigma : \mathbb{R} \to \mathbb{R}$ is a nonlinear activation function. For a vector $\mathbf{x}$, the expression $\mathbf{v} = \sigma(\mathbf{x})$ denotes componentwise application: $v_i = \sigma(x_i)$.

The entries of the matrices $\mathbf{W}_i$ are referred to as the weights or parameters of the network. In this work, we assume all activation functions are ReLU, i.e., $\sigma(x) = \max(0, x)$. This assumption is made for simplicity; the results can be extended to general activation functions as discussed in Burkholz [2022a].

For a neural network $f(\mathbf{x}) = \mathbf{W}_\ell \sigma(\mathbf{W}_{\ell-1} \cdots \sigma(\mathbf{W}_1 \mathbf{x}))$, we refer to the quantity $\sigma(\mathbf{W}_k \cdots \sigma(\mathbf{W}_1 \mathbf{x}))$ as the output of the k^{th} layer.

We next define some quantization strategies for neural networks which capture mixed-precision quantization practices. We defer the reader to the quantization paragraph in the Related Work (Section 2) for a discussion of such practices.

Definition 4. *A* δ-quantized neural network *is a neural network whose weights are sampled uniformly from the set* $S_\delta = \{-1, \ldots, \delta, \ldots, 1\}$, *where* $\delta = 2^{-k}$ *for some* $k \in \mathbb{N}$.

Definition 5. *A neural network* f *is called a* γ-double mixed precision neural network *if the output of each layer is quantized to precision* γ*, i.e.,*

$$f(\mathbf{x}) = [\mathbf{W}_\ell[\sigma(\mathbf{W}_{\ell-1} \cdots [\sigma(\mathbf{W}_1 \mathbf{x})]_\gamma)]_\gamma]_\gamma.$$

Definition 6. *A neural network* f *is called an* γ-triple mixed precision neural network *if the outputs of its even-numbered layers are quantized to precision* γ*, i.e.,*

$$f(\mathbf{x}) = [\mathbf{W}_{2\ell} \sigma(\mathbf{W}_{2\ell-1} \cdots [\sigma(\mathbf{W}_2(\sigma(\mathbf{W}_1 \mathbf{x})))]_\gamma)]_\gamma.$$

More generally, a *mixed-precision neural network* may reset the precision to γ at some layers while leaving others unquantized. Reducing the precision of a δ-quantized neural network f to γ means all weights of f are mapped to $[\cdot]_\gamma$. We denote this operation as $[f]_\gamma$.

Our objective is to represent a *target* Double Mixed Precision neural network f, with weights which are δ_1 quantized, using a second, potentially overparameterized, mixed-precision network g with finer quantization δ_2, by quantizating and pruning it. For a neural network

$$g(\mathbf{x}) = \mathbf{M}_{2\ell} \sigma(\mathbf{M}_{2\ell-1} \cdots \sigma(\mathbf{M}_1 \mathbf{x})).$$

the pruned network $g_{\mathbf{S}_i}$ is defined as:

$$g_{\{\mathbf{S}_i\}}(\mathbf{x}) = (\mathbf{S}_{2\ell} \odot \mathbf{M}_{2\ell}) \sigma((\mathbf{S}_{2\ell-1} \odot \mathbf{M}_{2\ell-1}) \cdots \sigma((\mathbf{S}_1 \odot \mathbf{M}_1)\mathbf{x})),$$

where each $\mathbf{S}_i$ is a binary pruning mask with the same dimensions as $\mathbf{M}_i$, and $\odot$ denotes element-wise multiplication. The goal is to find masks $\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_\ell$ such that f can be represented by the quantized and pruned version of g.

4.2 Quantized SLTH Results

Having discussed the pervious work on NPP and it's connection to RSSP, we now apply these results to prove results on SLTH in quantized setting. The main question that we want to answer is the following: Suppose we are given a target neural network, whose weights are of precision δ_t and a large network whose weights are of precision δ_{in}, such that $\delta_t \geq \delta_{\text{in}}$. Suppose we have the freedom to reduce the precision of the large network to δ, and then we can prune it. What is the relationship between δ and size of the large network such that the bigger network can be pruned to the target network. Now we state our first main result, which is analogs to the theorem proved by Pensia et al. [2020], but in the quantized setting.

Theorem 1. *Let* $\mathcal{F}$ *be the class of* δ_t *quantized* γ*-double mixed Precision neural networks of the form*

$$f(\mathbf{x}) = [\mathbf{W}_\ell[\sigma(\mathbf{W}_{\ell-1} \cdots [\sigma(\mathbf{W}_1 \mathbf{x})]_\gamma)]_\gamma]_\gamma.$$

Consider a 2ℓ *layered randomly initialized* δ_{in}*-quantized* γ*-touble mixed Precision neural network*

$$g(\mathbf{x}) = [\mathbf{M}_{2\ell} \sigma(\mathbf{M}_{2\ell-1} ... [\sigma(\mathbf{M}_2(\sigma(\mathbf{M}_1 \mathbf{x})))]_\gamma)]_\gamma,$$

with $\delta_{in}^2 \leq \delta_t$. *Let* $\delta_{in}^2 \leq \delta^2 \leq \delta_t$. *Assume* $\mathbf{M}_{2i}$ *has dimension*

$$d_i \times C d_{i-1} \log_2 \frac{1}{\delta},$$

and $\mathbf{M}_{2i-1}$ *has dimension*

$$Cd_{i-1}\log_2\frac{1}{\delta}\times d_{i-1}.$$

Then the precision of elements of $\mathbf{M}_i$*'s can be reduced to* δ*, such that for every* $f\in\mathcal{F}$*,*

$$\exists\ \{\mathbf{S}_i\}_{i=1}^{2\ell}:\quad [g_{\{\mathbf{S}_i\}}]_\delta(\mathbf{x})=f(\mathbf{x}).$$

with probability at least

$$1-N_t\ \mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right)$$

where N_t *is the total number of parameters in* f*.*

We prove the above theorem for a target network with a single weight (Lemma 5) using the results on RSSP in the previous section, and then we give the idea for proving it in general. The proof is an application of the strategy in Pensia et al. [2020] but with the use of Lemma 5. Details are given in Appendix B.

Lemma 5 (Representing a single weight)**.** *Let* $g:\mathbb{R}\to\mathbb{R}$ *be a randomly initialized* δ_{in} *quantized network of the form* $g(x)=[\mathbf{v}^T\sigma(\mathbf{u}x)]_\gamma$ *where* $\mathbf{u},\mathbf{v},\in\mathbb{R}^{2n}$*. Assume* $\delta_{in}^2\le\delta_t$ *and* $\delta_{in}^2\le\delta^2\le\delta_t$*. Also assume* $n>C\log_2\frac{1}{\delta}$*. Then the precision of weights of* g *can be reduced to* δ*, such that with probability atleast*

$$1-\mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right),$$

we have $\forall\ w\in S_{\delta_t}$

$$\exists\ \mathbf{s}^1,\mathbf{s}^2\in\{0,1\}^{2n}:[wx]_\gamma=[[(\mathbf{v}\odot\mathbf{s}^2)^T\sigma(\mathbf{u}\odot\mathbf{s}^1)]_\delta(\mathbf{x})]_\gamma.$$

Proof. Let the precision of g be δ. First decompose $wx=\sigma(wx)-\sigma(-wx)$. This is a general identity for ReLU non-linear activation and was introduced in Malach et al. [2020]. WLOG [2] say $w>0$. Let

$$\mathbf{v}=\begin{bmatrix}\mathbf{b}\\\mathbf{d}\end{bmatrix},\mathbf{u}=\begin{bmatrix}\mathbf{a}\\\mathbf{c}\end{bmatrix},\mathbf{s}^1=\begin{bmatrix}\mathbf{s}_1^1\\\mathbf{s}_2^1\end{bmatrix},\mathbf{s}^2=\begin{bmatrix}\mathbf{s}_1^2\\\mathbf{s}_2^2\end{bmatrix},$$

where $\mathbf{a},\mathbf{b},\mathbf{c},\mathbf{d}\in\mathbb{R}^n,\mathbf{s}_1^1,\mathbf{s}_2^1,\mathbf{s}_1^2,\mathbf{s}_2^2\in\{0,1\}^n$. This is shown diagrammatically in Figure 1 in Appendix D.

Step 1: Let $\mathbf{a}^+=\max\{\mathbf{0},\mathbf{a}\}$ be the vector obtained by pruning all the negative entries of $\mathbf{a}$. This is done by appropriately choosing $\mathbf{s}_1^1$ Since $w\ge 0$, then for all $x\le 0$ we have $\sigma(wx)=\mathbf{b}^T\sigma(\mathbf{a}^+x)=0$. Moreover, further pruning of $\mathbf{a}^+$ would not affect this equality for $x\le 0$. Thus we consider $x>0$ in next two steps. Therefore we get $\sigma(wx)=wx$ and $\mathbf{b}^T\mathbf{a}^+x=\sum_i b_ia_i^+x$.

Step 2: Consider the random variables $Z_i=b_ia_i^+$. These are numbers of precision δ^2, sampled from the set $\{ab\mid a,b\in S_\delta\}$. Now w, which is a number of precision δ_t, also belongs to the set $\{ab\mid a,b\in S_\delta\}$ because $\delta^2\le\delta_t$. The numbers Z_i's are not distributed uniformly, but by a standard rejection sampling argument (as in Lueker [1998]), there exists C such that more that $2\log_2\frac{1}{\delta}$ samples out of $C\log_2\frac{1}{\delta}$ are uniform distributed. We prune the other samples such that we are left with $\bar{Z}_i$, which are uniformly distributed. Now by Lemma 4, as long as cardinality of $\{\bar{Z}_i\}$ is greater than $2\log_2\frac{1}{\delta}$, the Random Subset Sum Problem with set $\{\bar{Z}_i\}$ and target w can be solved with probability atleast

$$p\ge 1-\mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right).$$

Note that solving the Subset Sum Problem in an integer setting where numbers are sampled from $\{-M,\dots,M\}$ and solving it when numbers are sampled from $\{-1,\dots\delta,2\delta,\dots,1\}$ is equivalent (only difference is a scaling factor). In Lemma 3 and 4, the sampling set is $\{-M,\dots,-1,1,\dots,M\}$, but 0 can be rejected during rejection sampling. Hence it follows that with probability p

$$\forall\ w\in S_\delta^+,\quad\exists\ \mathbf{s}_1\in\{0,1\}^n:w=\mathbf{b}^T\mathbf{s}_1\odot\mathbf{a}^+.$$

[2]Without Loss of Generality

where S_δ^+ denotes positive members of S_δ. The part shown in green in Figure 1 in Appendix D hence handles positive inputs.

Step 3: Similar to steps 1 and 2, we can prune negative weights from $\mathbf{c}$ and let the red part shown in Figure 1 in Appendix D handle negative inputs. It will follow that with probability p

$$\forall\ w \in S_\delta^+, \quad \exists\ \mathbf{s}_2 \in \{0,1\}^n : w = \mathbf{d}^T \mathbf{s}_2 \odot \mathbf{c}^-.$$

with probability p. Hence Lemma 4 follows. □

Proof Sketch for Theorem 1. The idea is to use the strategy follow the strategy in Pensia et al. [2020]. We represented a single weight in Lemma 5. Similarly we can represent a neuron by representing each of its weights (shown explicitly in Lemma 6 and diagrammatically in Figure 2 in Appendix B). Using the representation of a single neuron, we represent a full layer (shown explicitly in Lemma 7 and diagrammatically in Figure 3 in Appendix B). Then we represent a full network by applying Lemma 7 layer by layer. See Appendix B for details. □

Our next result employs construction from Burkholz [2022a].

Theorem 2. *Let $\mathcal{F}$ be the class of δ_t quantized γ-double mixed Precision neural networks of the form*

$$f(\mathbf{x}) = [\mathbf{W}_\ell[\sigma(\mathbf{W}_{\ell-1} \cdots [\sigma(\mathbf{W}_1\mathbf{x})]_\gamma)]_\gamma]_\gamma.$$

Consider an $\ell+1$ layered randomly initialized γ-mixed precision resetting network which resets the precession to γ in all layers except the first one,

$$g(\mathbf{x}) = [\mathbf{M}_{2\ell}\sigma[(\mathbf{M}_{2\ell-1}...[\sigma(\mathbf{M}_2(\sigma(\mathbf{M}_1\mathbf{x})))]_\gamma)]_\gamma]_\gamma,$$

whose weights are sampled from $\{-1\ldots,-\delta,\delta,\ldots,1\}$ with $\delta_{in} \leq \delta_t$. Let $\delta_{in} \leq \delta \leq \delta_t$. If $\mathbf{M}_1$ and $\mathbf{M}_2$ have dimensions

$$d_0 \times C d_0 \log_2 \frac{1}{\delta} \qquad \text{and} \qquad d_1 \log_2 \frac{1}{\delta} \times C d_0 \log_2 \frac{1}{\delta}$$

respectively, $\mathbf{M}_{i+1}$ has dimension greater than

$$d_i \log_2 \frac{1}{\delta} \times d_{i+1} \log_2 \frac{1}{\delta}$$

$\forall\ 2 < i < l-1$ and $M_{\ell+1}$ has dimension greater than

$$\log_2\left(\frac{1}{\delta}\right) d_{l-1} \times d_l.$$

Then the precision elements of $\mathbf{M}_i$'s can be reduced to δ such that for every $f \in \mathcal{F}$ we have

$$\exists\ \{\mathbf{S}_i\}_{i=1}^{\ell+1} : \quad [g_{\{\mathbf{S}_i\}}]_\delta(\mathbf{x}) = f(\mathbf{x}).$$

with probability atleast

$$1 - N_t\ \log_2\left(\frac{1}{\delta}\right)\ \mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right)$$

where N_t is the total number of parameters in f.

Proof Sketch for Theorem 2. We follow the construction in Burkholz [2022a]. The idea is to use the same trick as the previous result to represent a layer, but to copy it many times. Hence the representation of a layer which was supposed to give output $(x_1, x_2, \ldots, x_N)$, will give output $(x_1, x_1, \ldots, x_2, x_2, \ldots, x_N, x_N \ldots, x_N)$. These copies can now be used while representing the next layer, without adding an intermediate layer in between (shown in Lemma 8 and diagrammatically in Figure 4 in Appendix B). □

4.3 Lower Bound by Parameter Counting

In this section we show by a parameter counting argument, akin to that employed in Pensia et al. [2020], Natale et al. [2024], that there exists a two layered δ-quantized network with d^2 parameters that cannot be represented by a neural network with unless it has $\Omega\left(d^2 \log_2\left(\frac{1}{\delta}\right)\right)$ parameters. Note that any linear transformation $\mathbf{W}\mathbf{x}$ where $\mathbf{W} \in S_\delta^d \times S_\delta^d$ and $\mathbf{x} \in S_\delta^d$ can be expressed as a 2 layered neural network. Let $\mathcal{F}$ be the class of functions

$$\mathcal{F} := \{h_{\mathbf{W}} : \mathbf{W} \in S_\delta^d \times S_\delta^d\}, \qquad \text{where} \qquad h_{\mathbf{W}}(\mathbf{x}) = [\mathbf{I} \quad -\mathbf{I}]\, \sigma\left(\begin{bmatrix}\mathbf{W} \\ -\mathbf{W}\end{bmatrix}\mathbf{x}\right). \tag{2}$$

Theorem 3. *Let $g : \mathbb{R}^d \to \mathbb{R}^d$ be a δ quantized neural network of the form*

$$g(\mathbf{x}) = \mathbf{M}_\ell \sigma(\mathbf{M}_{\ell-1} ... \sigma(\mathbf{M}_1 \mathbf{x})),$$

where elements of $\mathbf{M}_i$'s are sampled from arbitrary distributions over S_δ. Let $\mathcal{G}$ be the set of all matrices that can be formed by pruning g. Let $\mathcal{F}$ be defined as in Eq. 2. If

$$\forall\, h \in \mathcal{F},\ \mathbb{P}\left(\exists\, g' \in \mathcal{G} : g' = h\right) \geq p,$$

then the total number of non zero parameters of g is at least

$$\log_2 p + d^2 \log_2\left(\frac{2}{\delta} + 1\right).$$

Proof Sketch. Theorem 3 follows from a parameter counting argument. We simply count the number of different functions in $\mathcal{F}$ and demand that with probability p, any $f \in \mathcal{F}$ be represented by pruning g. See Appendix B for details. □

The following immediate corollary of the previous theorem provide a matching lower bound to Theorem 1.

Corollary 1. *If g is a two-layer network satisfying the hypothesis of Theorem 3, then its width is $\Omega(d \log \frac{1}{\delta})$.*

5 Conclusion

We have proved *optimal* over–parameterization bounds for the Strong Lottery Ticket Hypothesis (SLTH) in the finite-precision setting. Specifically, we showed that any δ_t-quantized target network N_t can be recovered *exactly* by pruning a larger, randomly–initialized network N_{in} with precision δ_{in}. By reducing the pruning task to a *quantized* Random Subset Sum instance and importing the sharp phase-transition analysis for the Number Partitioning Problem, we derived width requirements that match the information-theoretic lower bound up to absolute constants. These results not only close the gap between upper and lower bounds for quantized SLTH, but also certify, for the first time, that pruning alone can yield *exact* finite-precision subnetworks rather than merely approximate ones. Beyond their theoretical interest, our findings pinpoint the precise interplay between quantization granularity and over-parameterization, and they suggest that mixed-precision strategies may enjoy similarly tight guarantees. An immediate open problem is to generalize our techniques to structured architectures—most notably convolutional, residual, and attention-based networks—where weight sharing and skip connections introduce additional combinatorial constraints. Another interesting direction is to incorporate layer-wise mixed precision and to analyze the robustness of lottery tickets under stochastic quantization noise, which is of interest for practical deployment on low-precision hardware accelerators. We believe that the combinatorial perspective adopted here will prove equally effective in these broader settings, ultimately advancing our theoretical understanding of extreme model compression.

References

Christian Borgs, Jennifer Chayes, and Boris Pittel. Phase transition and finite-size scaling for the integer partitioning problem. *Random Structures amp; Algorithms*, 19(3–4):247–288, October 2001. ISSN 1098-2418. doi: 10.1002/rsa.10004. URL http://dx.doi.org/10.1002/rsa.10004.

Rebekka Burkholz. Most activation functions can win the lottery without excessive depth. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022a. URL https://openreview.net/forum?id=NySDKS9SxN.

Rebekka Burkholz. Convolutional and Residual Networks Provably Contain Lottery Tickets. In *Proceedings of the 39th International Conference on Machine Learning*, pages 2414–2433, Baltimore, July 2022b. PMLR. URL https://proceedings.mlr.press/v162/burkholz22a.html.

Michael Carilli. Automatic Mixed Precision — PyTorch Tutorials 2.7.0+cu126 documentation, 2020. URL https://docs.pytorch.org/tutorials/recipes/recipes/amp_recipe.html.

James Diffenderfer and Bhavya Kailkhura. Multi-prize lottery ticket hypothesis: Finding accurate binary neural networks by pruning a randomly weighted network. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=U_mat0b9iv.

Damien Ferbach, Christos Tsirigotis, Gauthier Gidel, and Joey Bose. A General Framework For Proving The Equivariant Strong Lottery Ticket Hypothesis. In *The Eleventh International Conference on Learning Representations*, September 2022. URL https://openreview.net/forum?id=vVJZtlZB9D.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJl-b3RcF7.

Michael R Garey and David S Johnson. *Computers and intractability*. W.H. Freeman, New York, NY, April 1979.

Song Han, Huizi Mao, and William J. Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding, 2015. URL https://arxiv.org/abs/1510.00149.

Brian Hayes. The easiest hard problem. *American Scientist*, 90(2):113, 2002. ISSN 1545-2786. doi: 10.1511/2002.2.113. URL http://dx.doi.org/10.1511/2002.2.113.

Geonho Hwang, Yeachan Park, and Sejun Park. On expressive power of quantized neural networks under fixed-point arithmetic. *arXiv*, 2024. URL https://arxiv.org/abs/2409.00297.

George S. Lueker. On the Average Difference between the Solutions to Linear and Integer Knapsack Problems. In *Applied Probability-Computer Science: The Interface Volume 1*. Birkhäuser, 1982. ISBN 978-1-4612-5791-2. doi: 10.1007/978-1-4612-5791-2_22. URL https://dl.acm.org/doi/10.5555/313651.313692.

George S. Lueker. Exponentially small bounds on the expected optimum of the partition and subset sum problems. *Random Structures and Algorithms*, 12(1):51–62, January 1998. ISSN 1098-2418. URL http://dx.doi.org/10.1002/(SICI)1098-2418(199801)12:1<51::AID-RSA3>3.0.CO;2-S.

Eran Malach, Gilad Yehudai, Shai Shalev-Schwartz, and Ohad Shamir. Proving the lottery ticket hypothesis: Pruning is all you need. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6682–6691. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/malach20a.html.

Stephan Mertens. Phase transition in the number partitioning problem. *Physical Review Letters*, 81(20):4281–4284, November 1998. ISSN 1079-7114. URL http://dx.doi.org/10.1103/PhysRevLett.81.4281.

Marc Mézard and Andrea Montanari. *Information, Physics, and Computation*. Oxford University PressOxford, January 2009. ISBN 9780191718755. doi: 10.1093/acprof:oso/9780198570837.001.0001. URL http://dx.doi.org/10.1093/acprof:oso/9780198570837.001.0001.

Emanuele Natale, Davide Ferre', Giordano Giambartolomei, Frédéric Giroire, and Frederik Mallmann-Trenn. On the Sparsity of the Strong Lottery Ticket Hypothesis. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, November 2024. URL https://hal.science/hal-04741369v2.

Laurent Orseau, Marcus Hutter, and Omar Rivasplata. Logarithmic pruning is all you need. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, pages 2925–2934, Red Hook, NY, USA, December 2020. Curran Associates Inc. ISBN 978-1-7138-2954-6. URL https://proceedings.neurips.cc/paper/2020/file/1e9491470749d5b0e361ce4f0b24d037-Paper.pdf.

Ankit Pensia, Shashank Rajput, Alliot Nagle, Harit Vishwakarma, and Dimitris Papailiopoulos. Optimal lottery tickets via subset sum: Logarithmic over-parameterization is sufficient. In *Advances in Neural Information Processing Systems*, volume 33, pages 2599–2610, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1b742ae215adf18b75449c6e272fd92d-Paper.pdf.

Vivek Ramanujan, Mitchell Wortsman, Aniruddha Kembhavi, Ali Farhadi, and Mohammad Rastegari. What's hidden in a randomly weighted neural network?, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/papers/Ramanujan_Whats_Hidden_in_a_Randomly_Weighted_Neural_Network_CVPR_2020_paper.pdf.

Kartik Sreenivasan, Shashank Rajput, Jy-Yong Sohn, and Dimitris Papailiopoulos. Finding nearly everything within random binary networks. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 3531–3541. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/sreenivasan22a.html.

Yulong Wang, Xiaolu Zhang, Lingxi Xie, Jun Zhou, Hang Su, Bo Zhang, and Xiaolin Hu. Pruning from scratch, 2019. URL https://arxiv.org/abs/1909.12579.

Tim Dettmers Younes Belkada. A Gentle Introduction to 8-bit Matrix Multiplication for transformers at scale using transformers, accelerate and bitsandbytes, 2022. URL https://huggingface.co/blog/hf-bitsandbytes-integration.

Hattie Zhou, Janice Lan, Rosanne Liu, and Jason Yosinski. Deconstructing lottery tickets: Zeros, signs, and the supermask. In *Advances in Neural Information Processing Systems*, pages 2599–2610, 2019. URL https://arxiv.org/abs/1905.01067.

A NPP and RSSP Results

We start by stating the result by Borgs et al. [2001] on NPP. Define $I_{n,z}$ as

$$Z_{n,z} = 2^n I_{n,z} \times \begin{cases} 1 & \text{if } z = 0 \\ 2 & \text{if } z > 0. \end{cases} \tag{3}$$

Theorem 4. *Let $C_0 > 0$ be a finite constant, let $M = M(n)$ be an arbitrary function of n, let*

$$\gamma_n = \frac{1}{M\sqrt{2\pi n c_M}} \qquad \text{where} \qquad c_M = \mathbb{E}\left(\frac{X^2}{M^2}\right) = \frac{1}{3} + \frac{1}{2M} + \frac{1}{6M^2},$$

and let z and z' be integers. Then,

$$\mathbb{E}[I_{n,z}] = \gamma_n \left(\exp\left(-\frac{z^2}{2nM^2c_M}\right) + O(n^{-1})\right).$$

Furthermore

$$\mathbb{E}[I_{n,z}I_{n,z'}] = 2\gamma_n^2 \left(\exp\left(-\frac{z^2+(z')^2}{2nM^2c_M}\right) + \mathcal{O}\left(\frac{1}{n}\right) + \mathcal{O}\left(\frac{1}{n\gamma_n 2^n}\right)\right) + \frac{\gamma_n}{2^n}\left(\delta_{z+z',0} + \delta_{z-z',0}\right)\exp\left(-\frac{z^2+(z')^2}{2nM^2c_M}\right)$$

if z and z' are of the same parity, i.e., both odd or both even, while $\mathbb{E}[I_{n,z}I_{n,z'}] = 0$ if z and z' are of different parity.

Now using 4, we prove 1

Proof of Lemma 1. Consider $z \neq 0$. From Theorem 4 we have

$$\mathbb{E}[I_{n,z}] = \gamma_n \left(\exp\left(-\frac{z^2}{2nM^2c_M}\right) + \mathcal{O}(n^{-1})\right).$$

If we multiply by 2^{n+1} we get

$$\mathbb{E}[Z_{n,z}] = \rho_n \left(\exp\left(-\frac{z^2}{2nM^2c_M}\right) + \mathcal{O}(n^{-1})\right). \tag{4}$$

It also follows from the above equation that

$$\mathbb{E}[Z_{n,z}] \geq \rho_n \exp\left(-\frac{z^2}{2nM^2c_M}\right). \tag{5}$$

Furthermore, from Theorem 4 we have

$$\mathbb{E}[I_{n,z}I_{n,z'}] = 2\gamma_n^2 \left(\exp\left(-\frac{z^2+(z')^2}{2nM^2c_M}\right) + \mathcal{O}\left(\frac{1}{n}\right) + \mathcal{O}\left(\frac{1}{n\gamma_n 2^n}\right)\right) + \frac{\gamma_n}{2^n}\left(\delta_{z+z',0} + \delta_{z-z',0}\right)\exp\left(-\frac{z^2+(z')^2}{2nM^2c_M}\right)$$

If $z = z'$ we get

$$\mathbb{E}[I_{n,z}^2] = 2\gamma_n^2 \left(\exp\left(-\frac{2z^2}{2nM^2c_M}\right) + \mathcal{O}\left(\frac{1}{n}\right) + \mathcal{O}\left(\frac{1}{n\gamma_n 2^n}\right)\right) + \frac{\gamma_n}{2^n}\exp\left(-\frac{2z^2}{2nM^2c_M}\right)$$

Multiplying by $(2^{n+1})^2$ we get

$$\mathbb{E}[Z_{n,z}^2] = 2\rho_n^2 \left(\exp\left(-\frac{2z^2}{2nM^2c_M}\right) + \mathcal{O}\left(\frac{1}{n}\right) + \mathcal{O}\left(\frac{1}{n\rho_n}\right)\right) + 2\rho_n \exp\left(-\frac{2z^2}{2nM^2c_M}\right) \tag{6}$$

Now using Markov's inequality (Theorem 5, Appendix C) and Eq. 4 we get

$$\mathbb{P}(Z_{n,z} > 0) \leq \rho_n \left(\exp\left(-\frac{z^2}{2nM^2 c_M} \right) + \mathcal{O}\left(\frac{1}{n}\right) \right).$$

Using Cauchy-Schwartz inequality (Theorem 6, Appendix C) Eq. 5 and Eq. 6 we thus get

$$\mathbb{P}(Z_{n,z} > 0) \geq \frac{\rho_n^2 \exp\left(-\frac{2z^2}{2nM^2 c_M}\right)}{2\rho_n^2 \left(\exp\left(-\frac{2z^2}{2nM^2 c_M}\right) + \mathcal{O}\left(\frac{1}{n}\right) + \mathcal{O}\left(\frac{1}{n\rho_n}\right)\right) + 2\rho_n \exp\left(-\frac{2z^2}{2nM^2 c_M}\right)}$$

$$\Rightarrow \mathbb{P}(Z_{n,z} > 0) \geq \frac{1}{2\left(1 + \exp\left(\frac{z^2}{nM^2 c_M}\right)\left(\mathcal{O}\left(\frac{1}{n\rho_n}\right) + \mathcal{O}\left(\frac{1}{n}\right)\right) + \frac{1}{\rho_n}\right)}.$$

The same calculation can be done for $z = 0$, the only difference is that $Z_{n,z} = 2^n I_{n,z}$ (Eq. 3). □

Before moving ahead, lets establish the equivalence of NPP and SSP by proving Lemma 2.

Proof of Lemma 2. First of all notice that a NPP on the set $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ where X_i's are sampled uniformly from $\{-M, \ldots, -1, 1, \ldots, M\}$ can be solved iff the NPP on the set $\mathbf{X} = (|X_1|, |X_2|, \ldots, |X_n|)$ sampled uniformly from $\{1, 2, \ldots, M\}$ can be solved. This is because, first, it is obvious that $\{X_i\}_{i=1}^n$ is distributed uniformly over $\{1, 2, \ldots, M\}$, and secondly, the NPP does not care about the signs of the numbers, a sign can always be absorbed in the σ_i while solving the NPP.

We have an SPP with set $\mathbf{X}$, sampled uniformly from $\{-M, \ldots, -1, 1, \ldots, M\}$ and target t. Assume number partitioning problem can be solved, given the set $\mathbf{X}$ and target $\Lambda - 2t$. Notice that NPP does not care about the sign of the target, as an NPP with target k can be solved iff that NPP with target $-k$ can be solved. Assume there exists two partitions S_1 and S_2 of $\mathbf{X}$, with S_1 summing to x and S_2 summing to $\Lambda - x$, such that $\sum_{i \in S_2} i - \sum_{j \in S_1} j = (\Lambda - x) - x = \Lambda - 2$, which is equivalent to $\sum_{j \in S_1} j = x = t$. Hence, S_1 sums up to t, so the given SSP can be solved. The reverse direction also follows from the argument, proving the result. □

Proof of Lemma 3. Considers the number partitioning problem corresponding to the given random subset sum problem (Lemma 2). The target of this number partitioning problem is $z = \Lambda - 2t$. Consider $z \neq 0$. A key observation here is if Λ is even (event denoted by $\mathscr{E}_n$), then z is also even and if Λ is odd (event denoted by $\mathscr{O}_n$), then z is also odd. The probability that the random subset sum problem can be solved can be written in terms of the probability that the number partitioning problem can be solved

$$\mathbb{P}(Y_{n,t} > 0) = \mathbb{P}(\mathscr{E}_n)\mathbb{P}(Z_{n,z} > 0|\mathscr{E}_n) + \mathbb{P}(\mathscr{O}_n)\mathbb{P}(Z_{n,z} > 0|\mathscr{O}_n)$$

Since on $\mathscr{E}_n$, z is always even and on $\mathscr{O}_n$, z is always odd, we have two cases. If z is even, then

$$\mathbb{P}(Z_{n,z} > 0) = \mathbb{P}(\mathscr{E}_n)\mathbb{P}(Z_{n,z} > 0|\mathscr{E}_n).$$

If z is odd, then

$$\mathbb{P}(Z_{n,z} > 0) = \mathbb{P}(\mathscr{O}_n)\mathbb{P}(Z_{n,z} > 0|\mathscr{O}_n).$$

Hence $\mathbb{P}(Y_{n,t} > 0)$ can be written as

$$\mathbb{P}(Y_{n,t} > 0) = 2\mathbb{P}(Z_{n,z} > 0).$$

From Lemma 1 it follows that

$$\mathbb{P}(Y_{n,t} > 0) \leq 2\rho_n \left(\exp\left(-\frac{z^2}{2nM^2 c_M} \right) + \mathcal{O}\left(\frac{1}{n}\right) \right),$$

$$\mathbb{P}(Y_{n,t} > 0) \geq \frac{1}{\left(1 + \exp\left(\frac{z^2}{nM^2 c_M}\right)\left(\mathcal{O}\left(\frac{1}{n\rho_n}\right) + \mathcal{O}\left(\frac{1}{n}\right)\right) + \frac{1}{\rho_n}\right)}.$$

Same can be done for $z = 0$, only difference is a factor of 2. □

Proof of Lemma 4. We are given that $\lim_{n\to\infty}\kappa_n$ exists and is less than 1. Consider a more sensitive parametrization

$$\kappa_n = 1 - \frac{\log_2 n}{2n} + \frac{\lambda_n}{n} \qquad \text{or} \qquad M = \frac{2^{n+\lambda_n}}{\sqrt{n}}.$$

In this parametrization $\lim_{n\to\infty}\kappa_n < 1$ means $\lim_{n\to\infty}\lambda_n \to -\infty$. Note that in this regime $\rho_n \to \infty$. Now we have

$$\mathbb{P}(Y_{n,t} > 0) \geq \frac{1}{\left(1+\exp\left(\frac{(\Lambda-2t)^2}{nM^2c_M}\right)\left(\mathcal{O}\left(\frac{1}{n\rho_n}\right)+\mathcal{O}\left(\frac{1}{n}\right)\right)+\frac{1}{\rho_n}\right)}.$$

Now $t = \mathcal{O}(M)$ and demand Λ as

$$\Lambda < \frac{1}{\sqrt{3+\beta}} M\sqrt{n\log n}.$$

According to Hoeffding's inequality (Theorem 7, Appendix C), that happens with probability

$$\mathbb{P}\left(\Lambda < \frac{1}{\sqrt{3+\beta}} M\sqrt{n\log n}\right) \geq 1 - \exp\left(\frac{\frac{1}{3+\beta}M^2 n\log n}{4nM^2}\right)$$
$$\Rightarrow \mathbb{P}\left(\Lambda < \frac{1}{\sqrt{3+\beta}} M\sqrt{n\log n}\right) \geq 1 - \frac{1}{n^{\frac{1}{2(3+\beta)}}}.$$

Now as $\rho_n \to \infty$, we have

$$\mathbb{P}(Y_{n,t} > 0) \geq 1 - \mathcal{O}\left(\frac{1}{n^{\frac{\beta}{3+\beta}}}\right).$$

Hence the probability (say $\mathbb{P}(E)$) of events $P(Y_{n,t} > 0)$ and $\Lambda < \frac{1}{\sqrt{3+\beta}}M\sqrt{n\log n}$ happening together is given by

$$\mathbb{P}(E) = \left(1 - \mathcal{O}\left(\frac{1}{n^{\frac{\beta}{3+\beta}}}\right)\right)\left(1 - \frac{1}{n^{\frac{1}{2(3+\beta)}}}\right).$$

Note that this probability will converge to 1 fastest if $\beta = \frac{1}{2}$. Hence we choose $\beta = \frac{1}{2}$ and we get

$$\mathbb{P}(E) = \left(1 - \mathcal{O}\left(\frac{1}{n^{\frac{1}{7}}}\right)\right).$$

□

B SLTH-Quantization Results

In this appendix, we prove the results related to SLTH and weight quantization. We start by proving Theorem 1. The idea is to follow the strategy of Pensia et al. [2020], but use Lemma 5.

Lemma 6 (Representing a single Neuron)**.** *Consider a randomly initialized δ_{in} quantized neural network of the form $g(\boldsymbol{x}) = [\mathbf{v}^T\sigma(\mathbf{M}\boldsymbol{x})]_\gamma$ with $\boldsymbol{x} \in \mathbb{R}^d$. Assume $\delta_{in}^2 \leq \delta_t$ and $\delta_{in}^2 \leq \delta^2 \leq \delta_t$. Let $f_{\mathbf{w}}(\boldsymbol{x}) = [\mathbf{w}^T\boldsymbol{x}]_\gamma$ be a single layered δ_t quantized network. Let $\mathbf{M} \in \mathbb{R}^{Cd\log_2\frac{1}{\delta}\times d}$ and $\mathbf{v} \in \mathbb{R}^{Cd\log_2\frac{1}{\delta}}$. Then the precision of weights of g can be reduced to δ, such that with probability atleast*

$$1 - d\,\mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right),$$

we have

$$\forall \mathbf{w} \in S_\delta^d \;\; \exists \;\; \mathbf{s}, \mathbf{T}: \;\; f_{\mathbf{w}}(\boldsymbol{x}) = [g_{\{\mathbf{s},\mathbf{T}\}}]_\delta(\mathbf{x}),$$

where $[g_{\{\mathbf{s},\mathbf{T}\}}]_\delta(\mathbf{x})$ is the pruned network for a choice of binary vector $\mathbf{s}$ and matrix $\mathbf{T}$, *Proof.* Assume weights of g are of precision δ. We prove the required results by representing each weight of the neuron using Lemma 5 (See Figure 2, Appendix D).

Step 1: We first prune $\mathbf{M}$ to create a block-diagonal matrix $\mathbf{M}'$. Specifically, we create M by only keep the following non-zero entries:

$$\begin{bmatrix} \mathbf{u}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{u}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{u}_d \end{bmatrix}, \quad \text{where } \ \mathbf{u}_i \in \mathbb{R}^{C \log_2 \frac{1}{\delta}}.$$

We choose the binary matrix $\mathbf{T}$ to be such that $\mathbf{M}' = \mathbf{T} \odot \mathbf{M}$. We also decompose $\mathbf{v}$ and $\mathbf{s}$ as

$$\mathbf{s} = \begin{bmatrix} \mathbf{s}_1 \\ \mathbf{s}_2 \\ \vdots \\ \mathbf{s}_d \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \\ \vdots \\ \mathbf{v}_d \end{bmatrix}, \quad \text{where } \ \mathbf{s}_i, \mathbf{v}_i \in \mathbb{R}^{C \log_2 \frac{1}{\delta}}.$$

Step 2: Consider the event

$$E_i : \ \ [w_i x_i] = [(\mathbf{v}_i \odot \mathbf{s}_i)^T \sigma(\mathbf{u_i} x_i)].$$

According to Lemma 5, this event happens with probability

$$p = 1 - \mathcal{O}\left(\left(\log_2 \frac{1}{\delta}\right)^{-\frac{1}{7}}\right).$$

The event (say E) in the assumption of Lemma 6 corresponds with the intersection of these events $E = \cap_{i=1}^{d} E_i$. By taking a union bound (Theorem 8, Appendix C), E happens with a probability $dp - (d-1)$, which is equal to

$$1 - d\,\mathcal{O}\left(\left(\log_2 \frac{1}{\delta}\right)^{-\frac{1}{7}}\right).$$

The process is illustrated in Figure 2. Note that we want $> \log_2(\frac{1}{\delta})$ samples to be assured that a RSSP is solved with high probability, but we include that in the constant C. Any extra factors (a factor of 2 for example) is also absorbed in C throughout the proof. □

Lemma 7 (Representing a single layer)**.** *Consider a randomly initialized δ_{in} quantized two layer neural network of the form $g(\mathbf{x}) = [\mathbf{N}\sigma(\mathbf{M}\mathbf{x})]_\gamma$ with $x \in \mathbb{R}^{d_1}$. Assume $\delta_{in}^2 \leq \delta_t$ and $\delta_{in}^2 \leq \delta^2 \leq \delta_t$. Let $f_{\mathbf{W}}(\mathbf{x}) = [\mathbf{W}\mathbf{x}]_\gamma$ be a single layered δ_t quantized network. Assume $\mathbf{N}$ has dimension $d_2 \times C d_1 \log_2 \frac{1}{\delta}$ and $\mathbf{M}$ has dimension $C d_1 \log_2 \frac{1}{\delta} \times d_1$. Then the precision of weights of g can be reduced to to δ, such that with probability atleast*

$$1 - d_1 d_2\,\mathcal{O}\left(\left(\log_2 \frac{1}{\delta}\right)^{-\frac{1}{7}}\right),$$

$$\forall\ \mathbf{W} \in S_{\delta_1}^{d_1 \times d_2} \ \ \exists\ \ \mathbf{S}, \mathbf{T} : \ f_{\mathbf{W}}(\mathbf{x}) = [g_{\{\mathbf{S},\mathbf{T}\}}]_\delta(\mathbf{x}),$$

where $[g_{\{\mathbf{S},\mathbf{T}\}}]_\delta(\mathbf{x})$ is the pruned network for a choice of pruning matrices $\mathbf{S}$ and $\mathbf{T}$.

Proof. Assume weights of g are of precision δ. We first prune $\mathbf{M}$ to get a block diagonal matrix $\mathbf{M}'$

$$\mathbf{M}' = \begin{bmatrix} \mathbf{u}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{u}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{u}_{d_1} \end{bmatrix}, \quad \text{where } \ \mathbf{u}_i \in \mathbb{R}^{C \log_2 \frac{1}{\delta}}.$$

Thus, $\mathbf{T}$ is such that $\mathbf{M}' = \mathbf{T} \odot \mathbf{M}$. We also decompose $\mathbf{N}$ and $\mathbf{S}$ as following

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}_{1,1}^T & \cdots & \mathbf{s}_{1,d_1}^T \\ \mathbf{s}_{2,1}^T & \cdots & \mathbf{s}_{2,d_1}^T \\ \vdots & \ddots & \vdots \\ \mathbf{s}_{d_2,1}^T & \cdots & \mathbf{s}_{d_2,d_1}^T \end{bmatrix}, \quad \mathbf{N} = \begin{bmatrix} \mathbf{v}_{1,1}^T & \cdots & \mathbf{v}_{1,d_1}^T \\ \mathbf{v}_{2,1}^T & \cdots & \mathbf{v}_{2,d_1}^T \\ \vdots & \ddots & \vdots \\ \mathbf{v}_{d_2,1}^T & \cdots & \mathbf{v}_{d_2,d_1}^T \end{bmatrix}, \quad \text{where } \mathbf{v}_{i,j}, \mathbf{s}_{i,j} \in \mathbb{R}^{C \log_2 \frac{1}{\delta}}.$$

Now note that pruning $\mathbf{u}_i$ and $\mathbf{v}_{i,j}$ (using $\mathbf{s}_{i,j}$) is equivalent to Lemma 6. Hence it's simply an application of Lemma 5 d_1d_2 times. Hence the event in assumption of Lemma 7 occurs with a probability $d_1d_2p - (d_1d_2 - 1)$, by a union bound (Theorem 8, Appendix C), which is equal to

$$1 - d_1d_2\,\mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right).$$

The process is illustrated in Figure 3, Appendix D. Note that we want $> \log_2(\frac{1}{\delta})$ samples to be assured that a RSSP is solved with high probability, but we include that in the constant C. Constant Factors also absorbed in C. □

Proof of Theorem 1. Now we can see that Theorem 1 can be proved by applying Lemma 7 layer wise, where two layers of the large network represent one layer of the target. Note that the precision is set of δ_1 after every layer (of the large network) and precision is set of δ_1 after every layer (of the target network). Let the total number of parameters in the target network be N_t, i.e.,

$$N_t = \sum_{i=1}^{l-1} d_i d_{i+1}.$$

Then the event in assumption of Theorem 1, by union bound (Theorem 8, Appendix C), occurs with a probability $N_tp - (N_t - 1)$, where which is equal to

$$1 - N_t\,\mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right).$$

□

This construction improves the depth

In this subsection, we adapt construction by Burkholz [2022a] to prove Theorem 2. The process is illustrated in Figure 4, Appendix D.

Lemma 8. *Consider a randomly initialized δ_{in} quantized two layered neural network $g(\mathbf{x}) = [\mathbf{N}\sigma(\mathbf{M}\mathbf{x})]_\gamma$ with $\mathbf{x} \in \mathbb{R}^{d_1}$, whose weights are sampled uniformly from $\{-1, \dots, -\delta, \delta, \dots, 1\}$. Assume $\delta_{in}^2 \leq \delta_t$ and $\delta_{in}^2 \leq \delta^2 \leq \delta_t$. Let*

$$f_{\mathbf{W}}(\mathbf{x}) = \begin{bmatrix} [\mathbf{W}\mathbf{x}]_\gamma \\ [\mathbf{W}\mathbf{x}]_\gamma \\ \vdots \\ [\mathbf{W}\mathbf{x}]_\gamma \end{bmatrix}$$

be a single layered δ_t quantized network where $\mathbf{W}\mathbf{x}$ is repeated $\log_2(\frac{1}{\delta})$ times and $\mathbf{W}$ has dimension $d_1 \times d_2$. If $\mathbf{N}$ has dimension $d_2\log_2\frac{1}{\delta} \times Cd_1\log_2\frac{1}{\delta}$ and $\mathbf{M}$ has dimension $Cd_1\log_2\frac{1}{\delta} \times d_1$. Then the precision of weights of g can be reduced to to δ, such that with probability

$$1 - d_1d_2\log_2\left(\frac{1}{\delta}\right)\,\mathcal{O}\left(\left(\log_2\frac{1}{\delta}\right)^{-\frac{1}{7}}\right).$$

we have

$$\forall\, \mathbf{W} \in S_\delta^{d_1\times d_2}\ \ \exists\ \ \mathbf{S},\mathbf{T}:\ f_{\mathbf{W}}(\mathbf{x}) = [g_{\{\mathbf{S},\mathbf{T}\}}]_\delta(\mathbf{x}),$$

where $[g_{\{\mathbf{S},\mathbf{T}\}}]_\delta(\mathbf{x})$ is the pruned network for a choice of pruning matrices $\mathbf{S}$ and $\mathbf{T}$.

Proof. Assume weights of g are of precision δ. We first prune $\mathbf{M}$ to get a block diagonal matrix $\mathbf{M}'$

$$\mathbf{M}' = \begin{bmatrix} \mathbf{u}_1 & 0 & \cdots & 0 \\ 0 & \mathbf{u}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{u}_{d_1} \end{bmatrix}, \qquad \text{where}\ \ \mathbf{u}_i \in \mathbb{R}^{C\log_2\frac{1}{\delta}}.$$

Thus, $\mathbf{T}$ is such that $\mathbf{M}' = \mathbf{T} \odot \mathbf{M}$. We also decompose $\mathbf{N}$ and $\mathbf{S}$ as following

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_1 \\ \mathbf{S}_2 \\ \vdots \\ \mathbf{S}_{\log_2\left(\frac{1}{\delta}\right)} \end{bmatrix} \qquad \mathbf{N} = \begin{bmatrix} \mathbf{N}_1 \\ \mathbf{N}_2 \\ \vdots \\ \mathbf{N}_{\log_2\left(\frac{1}{\delta}\right)} \end{bmatrix}$$

where

$$\mathbf{S}_k = \begin{bmatrix} (\mathbf{s}_{1,1}^T)_k & \cdots & (\mathbf{s}_{1,d_1}^T)_k \\ (\mathbf{s}_{2,1}^T)_k & \cdots & (\mathbf{s}_{2,d_1}^T)_k \\ \vdots & \ddots & \vdots \\ (\mathbf{s}_{d_2,1}^T)_k & \cdots & (\mathbf{s}_{d_2,d_1}^T)_k \end{bmatrix}, \qquad \mathbf{N} = \begin{bmatrix} (\mathbf{v}_{1,1}^T)_k & \cdots & (\mathbf{v}_{1,d_1}^T)_k \\ (\mathbf{v}_{2,1}^T)_k & \cdots & (\mathbf{v}_{2,d_1}^T)_k \\ \vdots & \ddots & \vdots \\ (\mathbf{v}_{d_2,1}^T)_k & \cdots & (\mathbf{v}_{d_2,d_1}^T)_k \end{bmatrix},$$

and $(\mathbf{v}_{i,j})_k, (\mathbf{s}_{i,j})_k \in \mathbb{R}^{C \log_2 \frac{1}{\delta}}$.

Now note that pruning $\mathbf{u}_i$ and $(\mathbf{v}_{i,j})_k$ (using $(\mathbf{s}_{i,j})_k$) is equivalent to Lemma 6. Hence it's simply an application of Lemma 5 $d_1 d_2 \log_2\left(\frac{1}{\delta}\right)$ times. Hence the event in assumption of Lemma 8 occurs with a probability

$$1 - d_1 d_2 \log_2\left(\frac{1}{\delta}\right) \mathcal{O}\left(\left(\log_2 \frac{1}{\delta}\right)^{-\frac{1}{7}}\right),$$

using the union bound (Theorem 8, Appendix C). □

Proof of Theorem 2. In Lemma 8 we represented the first layer of the target network, with a difference that output contains many copies. The rest of the proof is same is Burkholz [2022a]. These copies can be used to represent weights in the next layer. The argument follows iteratively for all layers until we reach the last layer, where copying is not required. The only key difference is that rejection sampling is not required, giving the required size free of any undetermined constants. The process is illustrated in Figure 3. The event in the assumption of Theorem 2 happens with probability

$$1 - N_t \; \log_2\left(\frac{1}{\delta}\right) \mathcal{O}\left(\left(\log_2 \frac{1}{\delta}\right)^{-\frac{1}{7}}\right)$$

□

Lower Bound by Parameter Counting

Here we prove Theorem 3 which follows by a parameter counting in the discrete setting.

Proof of Theorem 3. Two matrices represent the same function iff all their elements are the same. Therefore, the number of functions in $\mathcal{F}$ is

$$\left(\frac{2}{\delta} + 1\right)^{d^2}.$$

Let the number of non zero parameters in g be α, then the number of functions in $\mathcal{G}$ is 2^α. Now for the assumption of Theorem 3 to hold, we must have

$$2^\alpha \geq p\left(\frac{2}{\delta} + 1\right)^{d^2}$$
$$\implies \alpha \geq \log_2 p + d^2 \log_2\left(\frac{2}{\delta} + 1\right).$$

□

Corollary 1 is an immediate consequence of Theorem 3.

C Inequalities

Theorem 5. *For a non-negative, integer-valued random variable X we have*

$$\mathbb{P}(X > 0) \leq \mathbb{E}[X].$$

Theorem 6. *If $X > 0$ is a random variable with finite variance, then*

$$\mathbb{P}(X > 0) \geq \frac{(\mathbb{E}[X])^2}{\mathbb{E}[X^2]}.$$

Theorem 7. *Let $X_1, X_2, \ldots, X_n$ be independent random variables such that $a_i \leq X_i \leq b_i$ almost surely. Consider the sum of these random variables,*

$$S_n = X_1 + X_2 + \cdots + X_n.$$

Then Hoeffding's theorem states that, for all $t > 0$,

$$\mathbb{P}(S_n - \mathbb{E}(s_n) \geq t) \leq \exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right)$$

$$\mathbb{P}(|S_n - \mathbb{E}(s_n)| \geq t) \leq 2\exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right).$$

Theorem 8. *For any events $A_1, A_2, \ldots, A_n$ we have*

$$\mathbb{P}\left(\bigcap_{i=1}^{n} A_i\right) \geq \max\left(0,\ \sum_{i=1}^{n} \mathbb{P}(A_i) - (n-1)\right).$$

562 # D Figures

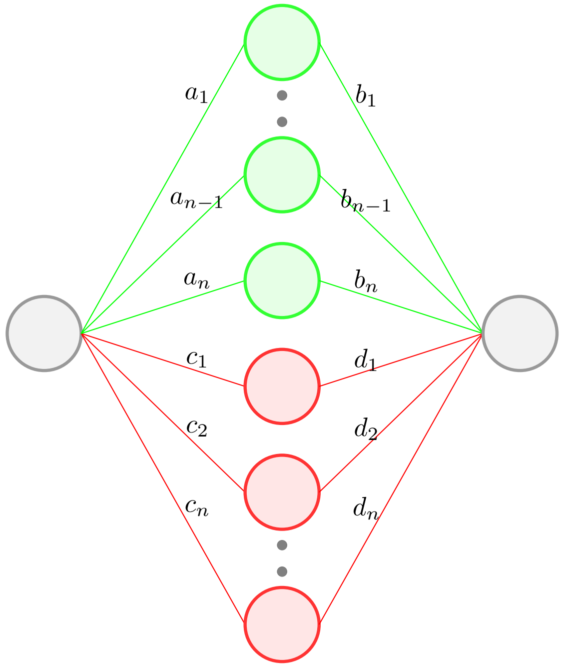

Figure 1: Approximating a single weight with ReLU activation (Pensia et al. [2020]): The network shown in the figure represents a single weight after pruning.

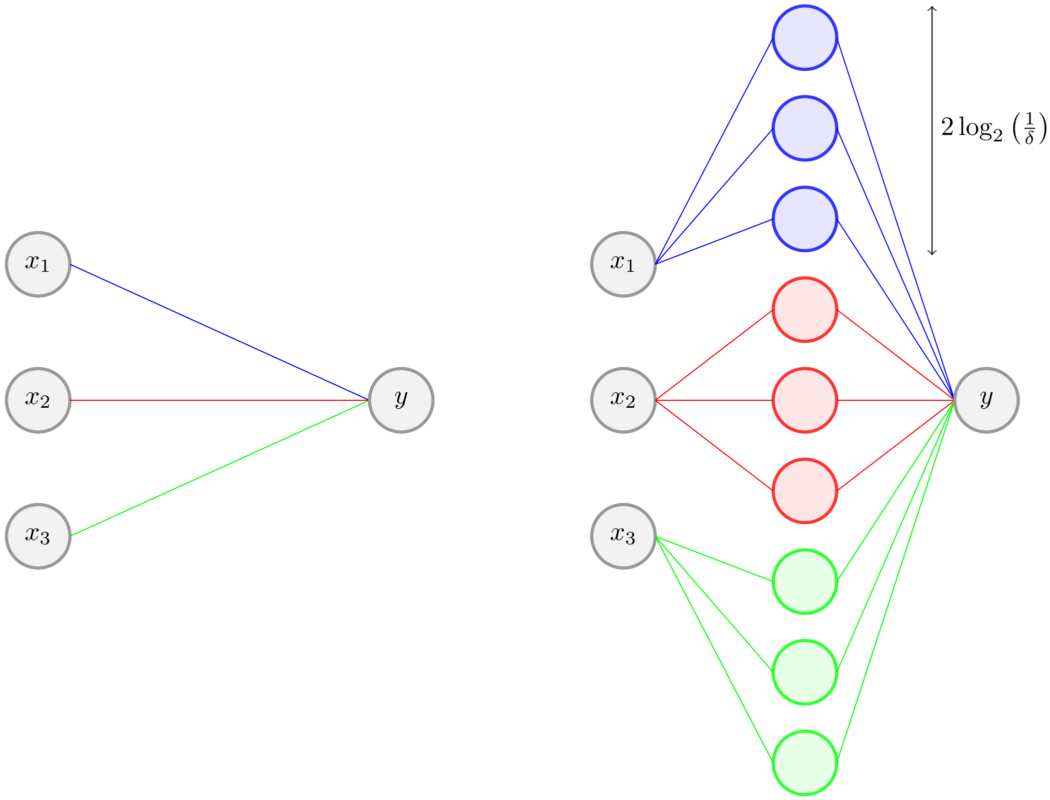

Figure 2: Representing a single neuron (Pensia et al. [2020]): The figure on the left shows the target network, where as Figure on the right shows the large network. The colors indicate which part in the target is represented by which part of the source. For example, the red weight on the left is represented by the red subnetwork on the right.

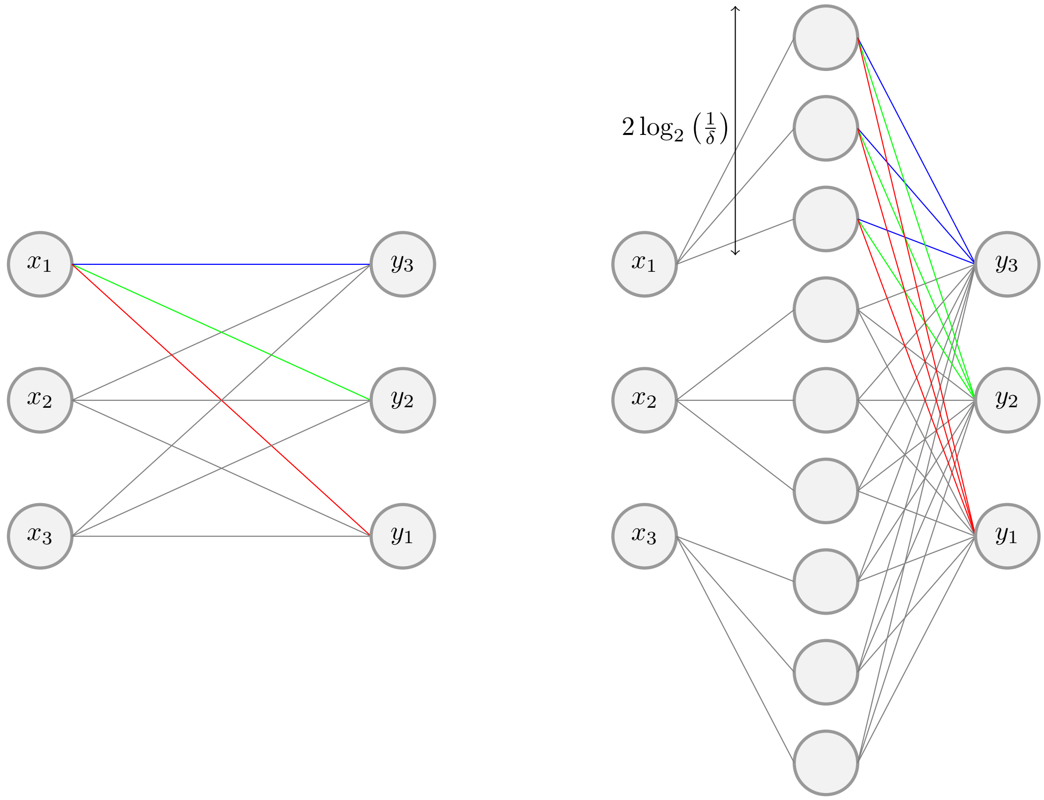

Figure 3: Representing a layer (Pensia et al. [2020]): Figure on the left shows the target network, where as Figure on the right shows the large network. The colors indicate which part in the target is represented by which part of the source. For example, the red weight on the left is represented by the red weights on the right.

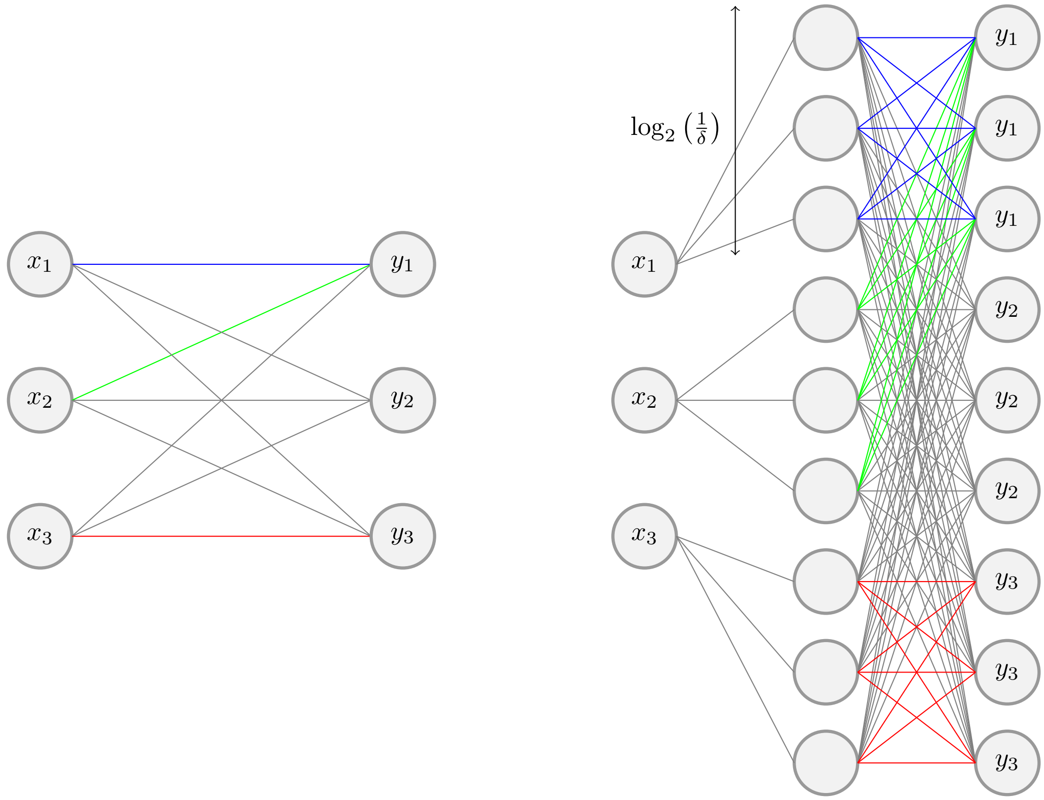

Figure 4: The Figure shows representation of first two layers of a network in Theorem 2: (Burkholz [2022a]). The figure on the left shows the target network, where as Figure on the right shows the large network. The colors indicate which part in the target is represented by which part of the source. For example, the red weight on the left is represented by the red weights on the right.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide complete proofs to the results stated in the abstract.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We sate our assumptions clearly, and we discuss where the assumptions might be unnatural, but were made for theoretical simplicity.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: We provide complete proofs to all the results in supplemental material. All Theorems are well numbered and cross referenced.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [NA]

Justification: The paper does not include experiments.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: The paper does not include experiments requiring code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [NA]

   Justification: The paper does not include experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: The paper does not include experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [NA]

   Justification: The paper does not include experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The authors have read the NeurIPS Code of Ethics and conform, in every respect with it.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: The paper does not use existing assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA] .

Justification: LLMs were used only for grammatical corrections in this work.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.