Применение условной оптимизации амплитуды шума квантования для низкобитного квантования глубоких нейросетей

Сергей Салищев

16 ноября 2023 г.

Аннотация

Основная цель работы – достичь глубокого теоретического понимания внутренней структуры и работы нейронных сетей (НС) как дискретного стохастического объекта. Вторая цель – достичь квантизации a1w2 или даже a1w1 как практического подтверждения первой цели. Третья цель – разработать практический метод квантования a2w2, a4w4 и a8w4 для сверточных классификаторов изображений и НС для преобразования изображения в изображение без потери качества по сравнению с вычислениями с плавающей точкой (FP) с использованием методов оптимизации с ограничениями и теории информации. Для достижения этих целей предлагается объединить три основных подхода: бинарное кодирование для входов и выходов НС, модификацию данных НС с помощью некоррелированного шума и преобразования НС для уменьшения разрядности битов.

1 Введение

Общая идея работы заключается в том, что нейронная сеть (НС) по своей природе является смесью дифференцируемых и стохастических дискретных систем. Хотя совершенно ясно, почему дифференцируемая часть системы может обучаться методом стохастического градиентного спуска (SGD), пока не до конца понятно, почему дискретная часть системы также может обучаться таким же образом.

Ключ к этому может находиться в области теории информации и, в частности, в теоремах о кодировании канала (Channel Coding Theorems, CCT). Обратим внимание, что человек может определить категорию предмета, например, стол по небольшой части его изображения. Если рассматривать процесс генерации входных данных НС, например изображений из параметров как канал, НС можно представить как декодер некоторого кода с коррекцией ошибок (FEC) для канала генерации входных данных. Если предположить, что взаимная информация между входом НС и скрытым представлением данных (bottleneck features) равна энтропии входных данных НС, то это представление содержит в себе и искомый код коррекции ошибок, а хвостовая часть НС после скрытого представления – это декодер этого кода. Если также рассматривать НС как канал, а информационная емкость скрытого представления НС выше, чем скорость кода коррекции ошибок для канала генерации входных данных, то, согласно CCT, любое случайное отображение входных данных на пространство скрытого представления будет также обладать свойствами кода коррекции ошибок с вероятностью $P = 1$. Случайность отображения достигается за счет стохастичности SGD.

Если рассматривать шум квантования НС как шум в канале, можно также применить теорему о кодировании для канала с аддитивным шумом шумом Хартли к самой НС, например, [RM14], что дает нам представление о пропускной способности квантованного канала для дифференцируемой части НС.

Поскольку НС вычисляется на бинарных цифровых схемах, то понтно, что ее можно бинаризовать без потери точности. Теория цифровых схем также предполагает, что при бинаризации без потерь глубина НС должна расти как функция от целочисленной разрядности вычислений

исходной НС. Объединяя два вышеуказанных соображения, можно сделать более сильное предположение, что можно эквивалентно по выходу преобразовать существующую НС в бинарную НС даже без точного сохранения промежуточных вычислений.

Практическая часть работы заключается в демонстрации рабочего метода квантования НС с преобразованием ее активаций в 1 бит без потери точности. Если это удастся, это будет свидетельством того, что представление о работе НС правильно. Промежуточные цели включают квантование с потерями и без потерь a2w2, a4w4 и a8w4, в соответствии с требованиями будущих аппаратных процессоров для тензорных вычислений (TPU), и достижение результатов, сопоставимых с существующими методами LSQ [EMB+19], LSQ+ [BLN+20], DAQ, QuantNoise, QSin [SCA+22].

Предложенный метод квантования состоит из следующих шагов:

1. Преобразование входных и выходных данных в 1 бит с помощью бинарных эмбеддингов.
2. Обучение эквивалентной НС с минимальной разрядностью.
3. Выполнение эквивалентного преобразования для уменьшения разрядности на самом широком слое.
4. Повторение шагов с шага 2 до достижения требуемой разрядности.

2 Теория информации в применении к НС

На основе работы Айтекина [Ayt22] и аналогичных исследований можно заключить, что все НС решают задачу классификации, так как любая регрессионная НС может быть построена на её основе. Рассмотрим НС прямого распространения (feed-forward) являющуюся идеальным классификатором:

$$c = f(x, \Theta) \tag{1}$$

$$h = w(c, Q), \quad h = IID \sim \mathbf{U} \tag{2}$$

$$x = v(h) + n_x \tag{3}$$

Здесь c — это класс, w — проекция класса на скрытое представление параметров реального мира, v — проекция из скрытого представления реального мира на наблюдения, n_x — шум наблюдений, усложняющий классификацию, Q — вектор скрытых параметров реального мира, Θ — вектор весов НС. Например, c может представлять класс «столы», что эквивалентно платоновской идее стола, а x — изображение конкретного стола. Все значения дискретны. Можно рассматривать h как код FEC, а f как декодер FEC.

Канал v является зашумлённым каналом. Согласно теоремам о кодировании канала без памяти с аддитивным некорректированным шумом, если пропускная способность канала v $C_v > N_c$, существует хороший код длиной N_h и длиной сообщения N_c, и любой случайный код будет хорошим с высокой вероятностью.

В большинстве приложений НС $f(x, \Theta)$ предполагается непрерывной и дифференцируемой, особенно при обучении Θ методом обратного распространения ошибки. Таким образом, f как канал имеет бесконечную пропускную способность. Реализация такой НС на основе вычислений с плавающей точкой будет избыточной по количеству вычислений.

Можно рассмотреть НС прямого распространения как композицию цепочки функций

$$\begin{aligned} x_1 &= f_1(x_0 + n_x, \Theta_n + n_\Theta) \\ &\dots \\ c = x_n &= f_n(x_{n-1} + n_x, \Theta_n + n_\Theta) \end{aligned} \tag{4}$$

$$n_x = IID \sim \mathbf{U}[-0.5, 0.5]^{k_i}, \quad n_\Theta = IID \sim \mathbf{U}[-0.5, 0.5]^{m_i} \tag{5}$$

Здесь n_Θ, n_x – шумы квантования в весах и активациях НС, которые распределены равномерно.

В этой формулировке каждую f_i можно также рассматривать как зашумлённый канал с шумом квантования. Используя теорему о кодировании канала Хартли с равномерным шумом, можно сделать вывод, что пропускная способность канала Хартли составляет

$$C_H = log_2(M) \tag{6}$$

где M – количество различимых уровней после округления. В этом контексте можно также рассматривать x_i как код FEC. Если пропускная способность канала $C_{f_i} > N_c$, существует хороший код длиной N_{x_i} и длиной сообщения N_c, и любой случайный код будет хорошим с высокой вероятностью. Этот подход коррелирует с наблюдением, что НС для задач классификации легко квантуются до низких бит. Это также может объяснить, почему обратное распространение ошибки работает даже после исчезновения градиента, так как SGD работает как случайное блуждание, и несложно найти хороший случайный код. Это также может объяснить, почему обучение НС иногда ухудшается при увеличении размера пакета (batch) по которому усредняется градиент.

3 Бинарные вложения (embeddings) для входа и выхода

Естественно предположить, что бинарная сеть должна иметь бинарные входы и выходы. Проблема с этим предположением заключается в том, что обучение НС затратно. Чтобы уменьшить работу по квантованию, желательно повторно использовать наилучшие из известных контрольных точек НС с плавающей точкой (FP), которые несовместимы с бинарными данными на входе. Можно предположить, что входы и выходы уже квантуются в некоторый целочисленный диапазон. Легко преобразовать целое число в битовое представление, что можно рассматривать как бинарное вложение для данных. Однако он не учитывает ключевое свойство упорядоченных данных, наличие расстояния. Расстояние между близкими точками должно эффективно рассчитываться бинарной НС, например, расстоянием Хэмминга.

Легко построить такое бинарное вложение $E : 2^n \to 2^{2^n-1}$, для которого расстояние Хэмминга будет точно соответствовать локальному расстоянию. Однако имеет экспоненциальную длину и не является эффективным.

$$E_k(p) = \begin{cases} 1, & p(2) > 2^k \\ 0, & \text{иначе} \end{cases} \tag{7}$$

$$p(x) = \sum_{k=0}^{n} p_k x^k \tag{8}$$

Оптимальное вложение специфично для задачи, например, для набора данных рукописных цифр MNIST изображения можно бинаризовать с самого начала без значительной потери точности. Наивное вложение выше слишком велико, поэтому нет смысла вставлять его искусственно.

Если вставить адаптерные слои перед и после сети и инициализировать их как тождественные, предложенная процедура обучит оптимальные вложения с соответствующими подсетями кодера и декодера как часть шага 3. Разрядности этих адаптерных слоёв не должны учитываться в разрядности НС.

Эту идею можно обобщить на n-битные вложения и использовать с другими методами квантования, такими как LSQ, LSQ+, для достижения равномерного квантования с улучшенной точностью.

Частичная реализация этого метода легко применимая на практике – отказ от квантования первого и последнего слоя НС.

4 Минимизация разрядности НС

Используя аргументацию Айтекина [Ayt22] об НС с любой нелинейностью как универсвльном аппроксиматоре для любой другой НС, без потери общности можем предположить, что нелинейность – это ReLU. Также предполагаем, что слои BatchNorm, если они используются, объединены с предыдущими линейными слоями. Сначала рассмотрим НС с плавающей точкой (FP).

$$x^{(i+1)} = ReLU(W^{(i)}[(x^{(i)})^T|1]^T) \tag{9}$$

$$ReLU(x) = max(0, x) = max_0 x \tag{10}$$

Здесь $x^{(i)}$ — активации, $x^{(1)}$ — вход НС, $x^{(N+1)}$ — выход НС с N слоями, $W^{(i)}$ — веса, включая смещение, а $[\ldots|\ldots]$ — горизонтальная конкатенация матриц. Это является объектом оптимизации параметров на обучающем наборе данных из M образцов

$$\Omega = [W_{jk}^{(i)}] \tag{11}$$

$$\Omega^* = argmin \sum_{k=1}^{M} L(x_k^{(1)}, x_k^{(N+1)}; \Omega) \tag{12}$$

Здесь Ω — вектор весов сети.

4.1 Общие положения

Рассмотрим сеть с целочисленными весами V и активациями y, которая «эквивалентна» НС с плавающей точкой (FP).

$$y^{(i+1)} = Q_a^{(i)}(ReLU(D_w^{(i)}(V^{(i)})D_a^{(i)}([(y^{(i)})^T|(D_a^{(i)})^{-1}(1)]^T))) \tag{13}$$

Здесь $Q^{(i)}$ — квантование, а $D^{(i)}$ — деквантование.

$$Q(x) = clamp(\lfloor s^{-1}(x-\beta) + 0.5 \rfloor, l, h), \quad l, h \in \mathbf{Z} \tag{14}$$

$$D(x) = sx + \beta \tag{15}$$

Эта НС неудобна для обучения методом обратного распространения ошибки (backward propagation), так как она не является дифференцируемой. Однако этот объект можно дифференцировать при добавлении некоррелированного шума с обучаемым параметром масштаба. Можно выразить $Q(x)$ с помощью аддитивного шума квантования:

$$Q(x) = s^{-1}\left(clamp(x, \hat{l}, \hat{h}) - \beta\right) + n(x), \quad \hat{l}, \hat{h} \in \mathbf{R} \tag{16}$$

$$n(x) \sim U([-0.5, 0.5]) \tag{17}$$

Здесь $n(x)$ — шум квантования, зависящий от x. Так как ∇n неизвестен, для дифференцируемости НС необходимо сделать шум независимым. Подобно LSQ+ используется асимметричное квантование для активаций и симметричное квантование для весов, чтобы они были на один бит шире. Для этого заменяем Q на $\bar{Q}_a$, y на $\bar{y}$ и V на $\bar{Q}_w(\mathcal{W})$. Для квантования весов принимаем:

$$\hat{l}_w^{(i)} = -max|\mathcal{W}_{jk}^{(i)}|, \hat{h}_w^{(i)} = max|\mathcal{W}_{jk}^{(i)}|, \beta_a = \beta_w = 0 \tag{18}$$

$$\bar{Q}_a(x) = s_a^{-1}(min_{\hat{h}}(max_{\hat{l}}x)) + u, \quad u = IID \sim U([-0.5, 0.5]) \tag{19}$$

$$D_a(\bar{Q}_a(x)) = min_{\hat{h}}(max_{\hat{l}}x) + s_a u_w \tag{20}$$

$$D_w(\bar{Q}_w(\mathcal{W})) = \mathcal{W} + s_w u_w \tag{21}$$

Зная, что нижняя граница для ReLU равна 0, можно упростить формулу $\hat{l}_a = 0, \hat{h}_a = q_a$. Пусть $x^{(i)} = D_a^{(i)}(\bar{y}^{(i)})$, тогда

$$x^{(i+1)} = min_{q_a^{(i)}}\left(ReLU\left((\mathcal{W}^{(i)} + s_w^{(i)}u_w^{(i)}\right)\left[(x^{(i)})^T|1\right]^T\right) + s_a^{(i)}u_a^{(i)} \tag{22}$$

Для нас min_q — это просто обратный ReLU с параметром:

$$min_q(x) = q - ReLU(q - x) \tag{23}$$

Теперь $x^{(i+1)}$ — дифференцируемая функция параметров:

$$x^{(i+1)} = f_\Theta\left(x^{(i)}, u_w^{(i)}, u_a^{(i)}\right), \quad u_w^{(i)} = IID \sim U([-0.5, 0.5])^{\cdots \times \cdots} \tag{24}$$

ПРИМЕЧАНИЕ: Использование $u_w^{(i)} = IID \sim U([-0.5, 0.5])^{n_i}$ плохо предсказывает округленные активации. Вместо этого используем предельное распределение шума округления, усреднённое по пакету. С большим пакетом результаты хорошие. Следует изучить фактическое распределение $n_a(x^i)$. Неформально, при вычислении ∇x некоррелированный шум усредняется в SGD, естественным образом приводя к Straight Through Estimator (STE) [BLC13]. Требуется более строгое обоснование для STE.

Можно определить функцию потерь, которая является p-нормой ширины квантования в битах:

$$\Theta = \left([\mathcal{W}^{(i)}], [s_w^{(i)}], [q_w^{(i)}][s_a^{(i)}], [q_a^{(i)}]\right), \quad s_w^{(i)}, q_w^{(i)}, s_a^{(i)}, q_a^{(i)} \in \mathbf{R}^+ \tag{25}$$

$$q_w^{(i)} = max(|\mathcal{W}^{(i)}|) \tag{26}$$

$$\mathcal{Q}(x^{(1)}, x^{(N+1)}; \Theta) = \sum \left| log_2 q_w^{(i)} - log_2 s_w^{(i)} \right|^p + \sum \left| log_2 q_a^{(i)} - log_2 s_a^{(i)} \right|^p \tag{27}$$

Количество бит для весов и активаций определяются как

$$b_w^{(i)} = \left\lceil log_2 Q_w^{(i)} (2^{q_w^{*(i)}} + 1) \right\rceil \tag{28}$$

$$b_a^{(i)} = \left\lceil log_2 Q_a^{(i)} (2^{q_a^{*(i)}}) \right\rceil \tag{29}$$

Теперь можно сформулировать задачу условной оптимизации:

$$\begin{cases} \Theta^* = argmin_\Theta \sum_k \mathcal{Q}(x_k^{(1)}, x_k^{(N+1)}; \Theta) \\ L(x_k^{(1)}, x_k^{(N+1)}; \Theta^*) \leq L(x_k^{(1)}, x_k^{(N+1)}; \Omega^*) \end{cases} \tag{30}$$

Последнее условие означает, что значение исходной функции потерь при квантовании не хуже, чем у FP НС. Следующий шаг — численно решить эту задачу условной оптимизации.

Чтобы быстро достичь практических результатов и оценить производительность метода по сравнению с другими методами, необходимо воспроизвести стандартную постановку задачи квантования для заданного целевого значения разрядности. Для этого формулируем двойную условную задачу оптимизации.

$$\begin{cases} \Omega^* = argmin \sum_{k=1}^{M} L(x_k^{(1)}, x_k^{(N+1)}; \Omega) \\ \left| log_2 q_w^{(i)} - log_2 s_w^{(i)} \right| \leq t_w^{(i)} \\ \left| log_2 q_a^{(i)} - log_2 s_a^{(i)} \right| \leq t_a^{(i)} \end{cases} \tag{31}$$

$$t_w^{(i)} = 2^{b_w^{(i)} - 1} - 1, \quad t_a^{(i)} = 2^{b_a^{(i)}} - 1 \tag{32}$$

Здесь $b_w^{(i)}, b_a^{(i)}$ — целевые разрядности для активаций и весов на каждом слое.

4.2 Решение задачи условной оптимизации для минимизации разрядности

Для численного решения задачи условной оптимизации используется метод множителей Лагранжа, который преобразует условную задачу в безусловную.

Для этого определяется новая функция потерь:

$$\mathcal{L}(x_k^{(1)}, x_k^{(N+1)}; \Theta, \alpha) = \mathcal{Q} + F(\alpha, L((x_k^{(1)}, x_k^{(N+1)}; \Theta), C(x_k^{(1)}, x_k^{(N+1)})) \tag{33}$$

$$\Theta^* = argmin_\Theta \sum_k \mathcal{L}(x_k^{(1)}, x_k^{(N+1)}; \Theta), \quad \alpha \to 0 \tag{34}$$

$$C(x_k^{(1)}, x_k^{(N+1)}) = L(x_k^{(1)}, x_k^{(N+1)}; \Omega^*) \tag{35}$$

Здесь F — функция штрафа, а C — функция ограничения на L. Существуют два распространённых типа штрафных функций:

1. Барьерная функция $F = -\alpha ln(C - L(\Theta) + \epsilon), \epsilon > 0$
2. Потенциальная функция $F = \alpha^{-1} max(0, C - L(\Theta))^p$

Были проверены обе функции — потенциальная и барьернная. Метод барьерной функции оказался неудачным из-за переполнения в барьере при использовании оптимизатора ADAM из-за использования момента (momentum). Поэтому далее рассматривается только метод потенциалов.

4.3 Применение метода потенциалов к минимизации разрядности

Окончательная формулировка задачи оптимизации:

$$\mathcal{L}(x_k^{(1)}, x_k^{(N+1)}; \Theta) = \mathcal{Q} + \alpha_r \alpha \left(max_0(C - L(\Theta)\right)^p \tag{36}$$

$$C = M^{-1} \sum_{k=1}^{M} L(x_k^{(1)}, x_k^{(N+1)}; \Omega^*) \tag{37}$$

$$\Theta^* = argmin \sum_k \mathcal{L}(x_k^{(1)}, x_k^{(N+1)}; \Theta) \tag{38}$$

Эта формулировка значительно упрощена по сравнению с предыдущей. Ограничение C является одним значением, чтобы исключить влияние FP HC из процесса; оно представляет собой гиперпараметр, означающий лучшую среднюю потерю на проверочном наборе данных. Параметры в $\mathcal{L}$ меняют масштаб на каждой итерации SGD, поэтому метод плохо сходится. Для улучшения сходимости применяется динамическая коррекция масштаба α_r, которая нормализует масштаб на каждой итерации SGD, основываясь на значениях предыдущих шагов. Гиперпараметр α является константой; хорошее эвристическое значение — $\alpha = 10$. Это типичная формулировка для оптимизации параметров HC, и она решается с помощью ADAM.

4.4 Квантование с целевой разрядностью

Используется метод потенциалов для решения соответствующей задачи условной оптимизации. Функция потерь определяется следующим образом:

$$\mathcal{L}(x_k^{(1)}, x_k^{(N+1)}; \Theta) = \bar{\mathcal{Q}} + \alpha^{-1} L(\Theta)^p \tag{39}$$

$$\Theta^* = argmin \sum_k \mathcal{L}(x_k^{(1)}, x_k^{(N+1)}; \Theta) \tag{40}$$

$$\begin{aligned} \bar{\mathcal{Q}}(x^{(1)}, x^{(N+1)}; \Theta) = \alpha_w \sum \left(max_0 \left(log_2 q_w^{(i)} - log_2 s_w^{(i)} - log_2 t_w^{(i)} \right) \right)^p \\ + \alpha_a \sum \left(max_0 \left(log_2 q_a^{(i)} - log_2 s_a^{(i)} - log_2 t_a^{(i)} \right) \right)^p \end{aligned} \tag{41}$$

$$t_w^{(i)} = 2^{b_w^{(i)} - 1} - 1, \quad t_a^{(i)} = 2^{b_w^{(i)}} - 1 \tag{42}$$

Здесь $b_w^{(i)}, b_a^{(i)}$ — целевые разрядности на каждом слое для активаций и весов, α_w, α_a — соответствующие динамические корректировки масштаба.

4.5 Квантование со смешанной разрядностью

В отличие от существующих методов, направленных на определённое значение разрядности, предлагаемый метод уменьшает разрядность постепенно. Это можно считать не недостатком, а важным преимуществом. Постепенное уменьшение разрядности по слоям при сохранении исходной функции потерь позволяет выявить слои, наиболее сильно влияющие на точность. Затем можно задать отдельную целевую разрядность для каждого слоя.

5 Преобразования для уменьшения разрядности

Можно явно уменьшить разрядность весов одного слоя, удвоив количество каналов и разделив этот слой, а также уменьшить разрядность активаций, удвоив количество каналов.

5.1 Уменьшение весов

Если квантизированная сеть имеет веса V^k с разрядностью $b_w^{(k)} \geq 2$, можно удвоить количество каналов k-го слоя с более низкой разрядностью весов и вставить рядом тривиальный слой слияния. Построим новую сеть с весами $\bar{V}^{(i)}$ и активациями $\bar{y}^{(i)}$.

1. Оставляем слои до k-го без изменений:

$$\bar{V}^{(i)} = V^{(i)}, \quad i < k \tag{43}$$

$$\bar{y}^{(i+1)} = y^{(i+1)}, \quad i < k \tag{44}$$

2. Дублируем каналы k-го слоя:

$$\bar{V}^{(k+1)} = (D_a^{(i)})^{-1}[I|I] \tag{45}$$

$$\bar{y}^{(k+1)} = \left[(y^{(i)})^T|(y^{(i)})^T\right]^T \tag{46}$$

3. Объединяем каналы в слое $(k+1)$:

$$\bar{V}^{(i)} = \left[V^+(i)|V^-(i)\right] \tag{47}$$

$$V^+(i) = \left\lceil \frac{V^{(i)}}{2} \right\rceil \tag{48}$$

$$V^-(i) = \left\lfloor \frac{V^{(i)}}{2} \right\rfloor \tag{49}$$

4. Увеличиваем индекс всех слоёв после k-го:

$$\bar{V}^{(i+1)} = V^{(i)}, \quad i > k \tag{50}$$

$$\bar{y}^{(i+2)} = y^{(i+1)}, \quad i > k \tag{51}$$

5.2 Уменьшение активаций

Аналогичное преобразование можно применить к активациям. Если квантизированная сеть имеет активацию $y^{(k)}$ с разрядностью $b_a^{(k)} \geq 2$, а $y^{(k-1)}$ имеет меньшую разрядность, можно разделить активации k-го слоя на два тензора с более низкой разрядностью. Если входы бинаризованы, предположение о меньшей разрядности $y^{(k-1)}$ выполняется для какого-то слоя, если разрядность активации $b_a^{(k)} \geq 2$.

Идея заключается в представлении $y^{(k+1)}$ как

$$y^{(k)} = y^{+(k)} + y^{-(k)}, \quad y^{+(k)}, y^{-(k)} \in \mathbf{R} \tag{52}$$

Зная, что нелинейность — ReLU, можем упростить Q_a:

$$Q_a(ReLU(x)) = \lfloor min_{\hat{h}} max_0(s^{-1}x) + 0.5 \rfloor, \quad \hat{h} \in \mathbf{R} \tag{53}$$

$$h^+ = \lceil \hat{h}/2 \rceil \tag{54}$$

$$h^- = \lfloor \hat{h}/2 \rfloor \tag{55}$$

$$y^{(k)} = \lfloor min_{\hat{h}} max_0(s^{-1}x) + 0.5 \rfloor \tag{56}$$

$$y^{+(k)} = \lfloor min_{h^+} max_0(s^{-1}x) + 0.5 \rfloor \tag{57}$$

$$y^{-(k)} = y^{(k)} - y^{+(k)} = \lfloor min_{h^-} max_0(s^{-1}x - y^{+(k)}) + 0.5 \rfloor \tag{58}$$

$$Q^+(x) = \lfloor min_{h^+}(s^{-1}x) + 0.5 \rfloor \tag{59}$$

$$Q^-(x) = \lfloor min_{h^-}(s^{-1}x) + 0.5 \rfloor \tag{60}$$

Тензор $y^{-(k)}$ является остатком, как в Residual NN. Этот трюк необходим, так как деление на 2 невозможно в целых числах. Также предполагается, что при обходе дополнительное квантование не используется. Построим новую сеть с весами $\bar{V}^{(i)}$ и активациями $\bar{y}^{(i)}$.

1. Оставляем все слои до $(k-1)$-го без изменений:

$$\bar{V}^{(i)} = V^{(i)}, \quad i < k \tag{61}$$

$$\bar{y}^{(i+1)} = y^{(i+1)}, \quad i < k \tag{62}$$

2. Разделяем $(k-1)$-й слой на $y^{+(k)}$ и обходной путь:

$$\bar{y}^{(k)} = [(y^{+(i+1))^T} | y^{(i))^T}]^T \tag{63}$$

$$\bar{Q}_a^{(k-1)} = Q^+ \tag{64}$$

3. Вставляем слой остатка и обходной путь:

$$\bar{y}^{(k+1)} = [(y^{+(i+1))^T} | y^{-(i))^T}]^T \tag{65}$$

$$\bar{Q}_a^{(k)} = Q^- \tag{66}$$

4. Объединяем $y^{-(k)}$ и $y^{+(k)}$:

$$\bar{V}^{(k+1)} = [V^{(k)} | V^{(k)}] \tag{67}$$

5. Увеличиваем индекс всех слоёв после $(k-1)$-го слоя:

$$\bar{V}^{(i+1)} = V^{(i)}, \quad i > k-1 \tag{68}$$

$$\bar{y}^{(i+2)} = y^{(i+1)}, \quad i > k-1 \tag{69}$$

6 Экспериментальные результаты

Большинство экспериментальных результатов получены с использованием 7-слойной CNN на датасете MNIST. Разделение слоёв пока не реализовано.

6.1 Нейронные сети для классификации

6.1.1 7-слойная CNN

7-слойная CNN состоит из пяти блоков, представленных как CONV-BN-RELU-MAXPOOL, одного слоя CONV в начале и одного слоя LINEAR в конце нейронной сети. HC во всех случаях использует 1-битные эмбеддинги для входа и выхода. Основная идея данной архитектуры — создать максимально простую архитектуру, чтобы достичь высокой точности на датасете MNIST при сохранении некоторого уровня сложности, чтобы продемонстрировать преимущества предложенного метода для квантования. Точность сохраняется вплоть до конфигурации квантования a3w4.

Таблица 1: Точность 7-слойной CNN на MNIST

Разрядность	Точность, %
FP	99.64
a3w4	99.65
a1w2	99.3

Таблица 2: Распределение разрядности для активаций 7-слойной CNN

Название активации	Разрядность
conv_block_1.noise.log_act_q	1.0
conv_block_2.noise.log_act_q	1.0
conv_block_3.noise.log_act_q	1.0
conv_block_4.noise.log_act_q	1.0
conv_block_5.noise.log_act_q	1.0

Таблица 3: Распределение разрядности для весов 7-слойной CNN

Название слоя весов	Разрядность
init_conv.conv2d	2.0
conv_block_1.conv2d	2.0
conv_block_2.conv2d	2.0
conv_block_3.conv2d	2.0
conv_block_4.conv2d	2.0
conv_block_5.conv2d	2.0
fc.lin	2.0

Как показано в таблицах 2 и 3, нам удалось достичь конфигурации a1w2 для всех слоёв в 7-слойной CNN, используя предложенный алгоритм. В отличие от рисунка 2, разрядность одинакова для всех слоёв и активаций благодаря использованию бинарных эмбеддингов для входа и выхода этой конкретной сети. ПРИМЕЧАНИЕ: Эти результаты достигнуты на предыдущей версии нашего фреймворка. Необходимо провести дополнительную работу для повторения их на текущей кодовой базе.

6.2 Нейронные сети для задач преобразования изображения в изображение

Была выбрана RFDN [LTW20] из-за небольшого количества параметров, что позволило быстрее проверять гипотезы. Датасетом для обучения был выбран Div2K, а валидационными датасетами — Set5, set14, b100, urban100.

ПРИМЕЧАНИЕ: предложенный метод применяется только к внутренним слоям RFDN, в то время как другие методы применяются ко всем слоям. Поэтому сравнение не является полностью корректным.

Основными метриками для сравнения методов квантования являются PSNR и SSIM, что делает сравнение более объективным способом оценить производительность подходов.

6.2.1 Сеть RFDN

Масштаб для сети RFDN был выбран X4 из-за наличия официального чекпоинта для этого масштаба и для достижения лучшей производительности.

В последних улучшениях удалось модифицировать предложенный подход для работы с конкретными разрядностями активаций и весов. Это позволило напрямую сравнить некоторые конфигурации разрядности с базовыми методами.

Объяснение этого поведения заключается в архитектуре RFDN. Как показано на рис. 1, существует остаточное соединение между первым и последним сверточными слоями, которые не квантованы, так что даже при низкой разрядности активаций сигнал всё равно проходит через нейронную сеть.

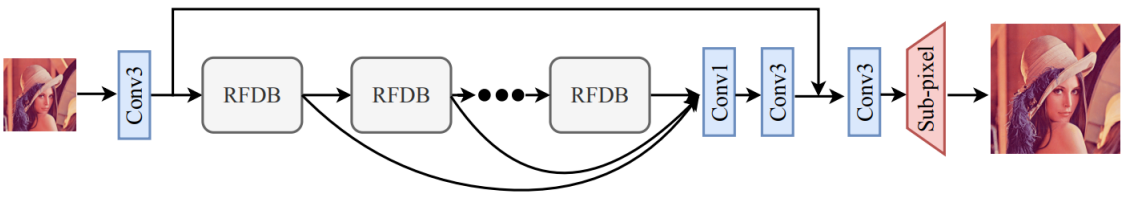

Рис. 1: Остаточное соединение в RFDN

6.3 Результаты квантования

В следующих таблицах представлено сравнение с конфигурацией a8w4 для передовых методов квантования, таких как LSQ, LSQ+, QSin.

Обучение RFDN с предложенным методом в течение 350 эпох заняло 25 минут и потребовало в общей сложности 3.1e15 FLOP во время обучения (без прохождения валидации).

Таблица 4: Метрики PSNR a8w4

	FP	LSQ	LSQ+	QSin	Предложенный метод
Set5	32.1280	26.8762	27.8236	29.2023	**30.5816**
Set14	27.9950	24.1806	24.9256	26.2054	**26.9312**
b100	27.4006	24.3306	24.9468	26.0671	**26.6396**
urban100	25.2399	21.5465	22.2563	23.3082	**23.5788**

Таблица 5: Метрики PSNR a4w4

	FP	LSQ	LSQ+	QSin	Предложенный метод
Set5	32.12804	26.87624	26.87178	26.55442	**29.73**
Set14	27.99506	24.18069	24.33979	24.30934	**26.34**
b100	27.40068	24.33064	24.56597	24.3421	**26.35**
urban100	25.23998	21.54658	21.61251	21.92647	**23.00**

Таблица 6: Метрики SSIM a8w4

	FP	LSQ	LSQ+	QSin	Предложенный метод
Set5	0.9084	0.8553	0.8538	0.8400	**0.8815**
Set14	0.8614	0.7984	0.8000	0.8194	**0.8398**
b100	0.7336	0.6853	0.6839	0.6644	**0.7075**
urban100	0.8869	0.8040	0.8056	0.8451	**0.8426**

Таблица 7: Метрики SSIM a4w4

	FP	LSQ	LSQ+	QSin	Предложенный метод
Set5	0.9084	0.7777	0.7822	0.7125	**0.8590**
Set14	0.8614	0.7479	0.7536	0.7144	**0.8274**
b100	0.7336	0.63152	0.6354	0.5815	**0.6974**
urban100	0.8869	0.7507	0.7610	0.7515	**0.8238**

Как видно на рисунке 2, нам удалось обучить нейронную сеть RFDN для достижения разрядности a4w4 во всех слоях, за исключением первого и последнего.

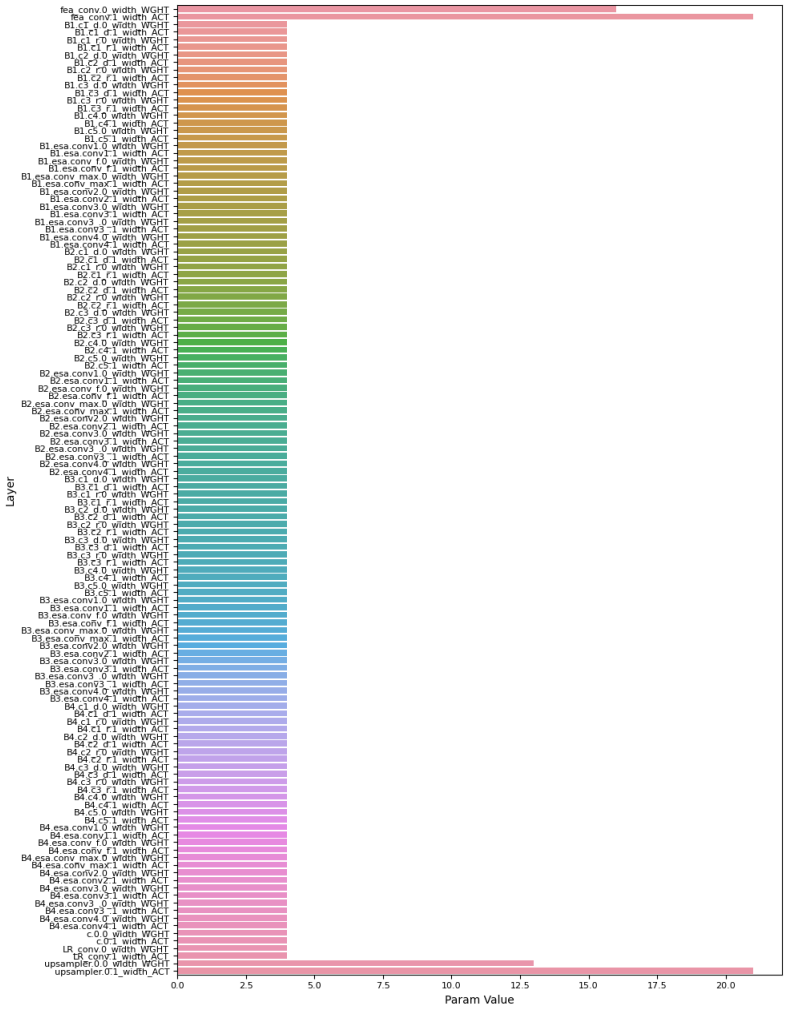

Рис. 2: Распределение разрядности для нейронной сети RFDN

6.4 Сравнение качества Set14 - Lenna

В этом разделе представлены результаты вывода квантованной сети. Изображения представлены в следующем порядке слева направо: исходное изображение, результат вывода сети с плавающей точкой без квантования, результат вывода квантованной сети с использованием предложенного метода в конфигурации a4w4, результат билинейного увеличения изображения. Можно увидеть, что квантованная НС хотя и генерирует небольшие блочные артефакты, но выдает существенно более резкое изображение по сравнению с интерполяцией.

Рис. 3: Set14 - 8 (GT, FP, Our a4w4, BL)

Благодарности

Автор выражает благодарность Яну Ахремчику и Николаю Пенкрату.

Список литературы

[Ayt22] Caglar Aytekin. Neural networks are decision trees. *arXiv preprint arXiv:2210.05189*, 2022.

[BLC13] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[BLN+20] Yash Bhalgat, Jinwon Lee, Markus Nagel, Tijmen Blankevoort, and Nojun Kwak. Lsq+: Improving low-bit quantization through learnable offsets and better initialization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 696–697, 2020.

[EMB+19] Steven K Esser, Jeffrey L McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dharmendra S Modha. Learned step size quantization. *arXiv preprint arXiv:1902.08153*, 2019.

[LTW20] Jie Liu, Jie Tang, and Gangshan Wu. Residual feature distillation network for lightweight image super-resolution. *arXiv preprint arXiv:2009.11551*, 2020.

[RM14] Olivier Rioul and José Carlos Magossi. On shannon's formula and hartley's rule: Beyond the mathematical coincidence. *Entropy*, 16(9):4892–4910, 2014.

[SCA+22] Kirill Solodskikh, Vladimir Chikin, Ruslan Aydarkhanov, Dehua Song, Irina Zhelavskaya, and Jiansheng Wei. Towards accurate network quantization with equivalent smooth regularizer. In *European Conference on Computer Vision*, pages 727–742. Springer, 2022.