УДК: 004.89

Худайберидева Г. Б., магистр, ассистент кафедры
«Информатика и информационные технологии»
Московский Политехнический Университет,
Россия, г. Москва

Кожухов Д. А., магистр, ассистент кафедры
«Информатика и информационные технологии»
Московский Политехнический Университет,
Россия, г. Москва

Пименкова А. А., студент-бакалавр кафедры
«Информатика и информационные технологии»
Московский Политехнический Университет,
Россия, г. Москва

СТАНДАРТИЗАЦИЯ И БЕЗОПАСНОЕ КОДИРОВАНИЕ ОБЪЕДИНЕНИЕ КВАНТОВАНИЯ, ПРУНИНГА И ДИСТИЛЛЯЦИИ В ЕДИНЫЙ АДАПТИВНЫЙ КОНВЕЙЕР ДЛЯ МИКРОКОНТРОЛЛЕРОВ КЛАССА CORTEX-M

***Аннотация**: Развертывание нейронных сетей на микроконтроллерах класса Cortex-M сопряжено с ограничениями по вычислительным ресурсам, объему памяти и энергопотреблению. Индивидуальное применение методов сжатия моделей, таких как квантование, прунинг и дистилляция знаний, демонстрирует ограниченную эффективность в условиях данных ограничений. Данная работа предлагает исследование синергетических эффектов при последовательном комбинировании указанных методов в едином адаптивном конвейере. Основное внимание уделяется анализу взаимозависимостей, например, влияния структурированного прунинга на последующее квантование. Предложена методология создания адаптивного инструмента, автоматически определяющего и настраивающего оптимальную последовательность и параметры методов сжатия для заданной целевой модели,*

целевого микроконтроллера Cortex-M и требуемых показателей точности. Экспериментальные результаты подтверждают, что предложенный адаптивный конвейер превосходит по эффективности изолированное применение методов сжатия, обеспечивая более высокую степень сжатия и ускорения при соблюдении целевых метрик точности на ресурсоограниченных устройствах.

***Ключевые слова**: сжатие нейронных сетей, квантование, прунинг, дистилляция знаний, адаптивный конвейер, микроконтроллеры Cortex-M, встраиваемые системы, ресурсоограниченные устройства, автоматизация оптимизации, TinyML.*

Khudaiberideva G. B.
master and department assistant at the department of
"Computer Science and Information Technology"
Moscow Polytechnic University
Moscow, Russia

Kozhukhov D. A.
master and department assistant at the department of
"Computer Science and Information Technology"
Moscow Polytechnic University
Moscow, Russia

Pimenkova A. A.
bachelor's student at the department of
"Computer Science and Information Technology"
Moscow Polytechnic University
Moscow, Russia

STANDARDIZATION AND SECURE CODING COMBINING QUANTIZATION, PRUNING, AND DISTILLATION INTO A SINGLE ADAPTIVE PIPELINE FOR CORTEX-M CLASS MICROCONTROLLERS

Annotation: *The deployment of neural networks on Cortex-M class microcontrollers is subject to limitations in computing resources, memory, and power consumption. Individual application of model compression methods such as quantization, pruning, and knowledge distillation demonstrates limited effectiveness under these constraints. This work suggests a study of synergetic effects when sequentially combining these methods in a single adaptive pipeline. The main focus is on the analysis of interdependencies, for example, the effect of structured pruning on subsequent quantization. A methodology is proposed for creating an adaptive tool that automatically determines and adjusts the optimal sequence and parameters of compression methods for a given target model, a target Cortex-M microcontroller, and required accuracy indicators. Experimental results confirm that the proposed adaptive pipeline is more efficient than the isolated application of compression methods, providing a higher degree of compression and acceleration while meeting the target accuracy metrics on resource-limited devices.*

Keywords: *neural network compression, quantization, pruning, knowledge distillation, adaptive pipeline, Cortex-M microcontrollers, embedded systems, resource-limited devices, optimization automation, TinyML.*

Введение

Актуальность развертывания моделей глубокого обучения на микроконтроллерах (МК) семейства Cortex-M, характеризующихся существенными ограничениями оперативной и постоянной памяти, тактовой частоты и энергопотребления, неуклонно возрастает в контексте развития Интернета вещей и периферийных вычислений [1]. Традиционные модели нейронных сетей обладают избыточной параметрической сложностью и вычислительными требованиями, делающими их прямое применение на МК класса Cortex-M непрактичным [2]. Для преодоления данных ограничений активно исследуются методы сжатия моделей, среди которых наиболее распространены квантование весов и активаций (Quantization), прунинг (Pruning) и дистилляция знаний (Knowledge Distillation) [3, 4]. Каждый из этих методов обладает уникальными характеристиками воздействия на модель: квантование

снижает битность представления данных, прунинг удаляет избыточные параметры или связи, дистилляция переносит знания от большой ("учитель") к малой ("ученик") модели. Однако изолированное применение данных методов часто не позволяет достичь необходимого баланса между степенью сжатия, скоростью вывода и сохранением точности для конкретных ограничений целевого МК [5]. Более того, порядок применения и параметры этих методов критически влияют на конечный результат, создавая сложную многомерную задачу оптимизации. Настоящая работа фокусируется на исследовании синергетических эффектов и взаимозависимостей при последовательном объединении квантования, прунинга и дистилляции в единый конвейер обработки и разработке адаптивной системы, автоматически конфигурирующей данный конвейер под специфические требования целевого устройства Cortex-M и приложения.

Постановка проблемы и анализ существующих подходов.

Проблема эффективного сжатия моделей для МК класса Cortex-M усугубляется их крайней гетерогенностью по вычислительной мощности (от M0 до M7), объему доступной памяти (десятки-сотни КБ ОЗУ, сотни КБ-единицы МБ ПЗУ) и поддержке аппаратных ускорителей (наличие/отсутствие SIMD инструкций, как в ARM CMSIS-NN) [6]. Существующие подходы к сжатию нейронных сетей можно условно разделить на методы, применяемые во время обучения (training-aware) и после обучения (post-training) [7]. К первым относится дистилляция знаний и обучение с учетом прунинга/квантования (QAT, Quantization-Aware Training). Ко вторым – посттренировочное квантование (PTQ, Post-Training Quantization) и посттренировочный прунинг. Преимущество посттренировочных методов заключается в меньшей вычислительной сложности и независимости от исходного процесса обучения модели [8]. Однако их эффективность, особенно в экстремальных условиях МК Cortex-

M, часто ниже, чем у методов, интегрированных в обучение. Ключевым недостатком большинства исследований и инструментов (таких как TensorFlow Lite Micro, STM32Cube.AI, Apache TVM) является применение методов сжатия по отдельности или в фиксированных, эмпирически подобранных комбинациях [9, 10]. Отсутствует систематический анализ того, как выбор типа прунинга (структурированный, неструктурированный, глобальный, поэлементный) влияет на последующее квантование весов и активаций, или как дистилляция может компенсировать потери точности, вызванные агрессивным прунингом и квантованием. Недостаточно изучен вопрос адаптации параметров конвейера сжатия (порядок методов, степень прунинга, битность квантования, архитектура модели-ученика для дистилляции) под конкретную целевую платформу Cortex-M с ее уникальными характеристиками и допустимым уровнем потери точности [11]. Существующие решения редко предоставляют инструменты для автоматизированного поиска оптимальной конфигурации конвейера сжатия, требуя от разработчика ручного перебора множества вариантов, что непрактично для сложных моделей.

Предлагаемый адаптивный конвейер сжатия.

Для решения обозначенных проблем предлагается единый адаптивный конвейер сжатия нейронных сетей, интегрирующий методы прунинга, квантования и дистилляции знаний. Инновационность подхода заключается в трех ключевых аспектах: исследование синергии методов, последовательное применение с учетом взаимовлияния и автоматизация выбора оптимальной конфигурации под целевую платформу. Конвейер функционирует как последовательность этапов обработки исходной модели.

Начальным этапом является структурированный прунинг. Приоритет отдается структурированным методам (удаление целых каналов, фильтров

или блоков), так как они обеспечивают предсказуемое уменьшение вычислительного графа модели, что критически важно для эффективной работы на МК с фиксированными аппаратными возможностями и оптимизированных библиотеках вроде CMSIS-NN [12]. Используются алгоритмы, основанные на оценке значимости параметров (например, по величине весов, по вкладу в активации или через анализ чувствительности слоев). Результатом этапа является модель с уменьшенной архитектурой.

Следующим этапом применяется квантование. Предлагается использовать посттренировочное квантование (PTQ) как менее ресурсоемкое, но при необходимости конвейер может быть расширен поддержкой квантовано-осознанного обучения (QAT) для более агрессивных настроек сжатия. Ключевой аспект заключается в том, что структура модели, полученная после прунинга, влияет на процесс калибровки квантования. Удаление избыточных каналов или фильтров может изменить распределение активаций в оставшихся слоях, что требует адаптивной настройки параметров квантования (диапазоны min/max, выбор схемы квантования – асимметричная, симметричная, per-channel/per-tensor) [13]. Результатом является модель с пониженной битностью весов и активаций (например, 8-битная, 4-битная).

Завершающим этапом конвейера является дистилляция знаний. Здесь модель, полученная после прунинга и квантования, выступает в роли "учителя". Цель этапа – восстановить точность, потенциально утраченную на предыдущих стадиях агрессивного сжатия, путем обучения компактной модели-"ученика" (возможно, с архитектурой, дополнительно оптимизированной под целевую платформу) имитировать выходы или внутренние представления "учителя" [14]. Использование сжатой модели в качестве "учителя" вместо исходной полной модели снижает вычислительные затраты на этапе дистилляции и позволяет

сфокусироваться на специфических особенностях уже оптимизированной модели.

Адаптивный механизм выбора конфигурации.

Сердцем предложенного подхода является адаптивный механизм, автоматически подбирающий оптимальную конфигурацию конвейера сжатия для заданных входных параметров: исходная модель, целевой микроконтроллер Cortex-M (с его спецификацией: тип ядра, объем RAM/Flash, наличие аппаратных ускорителей), целевой показатель точности (например, минимально допустимая Top-1 Accuracy). Данный механизм реализуется как система оптимизации с черным ящиком (black-box optimization). Пространство поиска включает: тип и степень агрессивности прунинга (процент удаляемых каналов/фильтров), битность квантования весов и активаций (возможны разные схемы для разных слоев), параметры дистилляции (температура, веса лоссов, архитектура модели-ученика), порядок применения методов (хотя базовый порядок Прунинг->Квантование->Дистилляция установлен как стартовый, механизм может исследовать вариации). В качестве целевой функции оптимизации выступает взвешенная комбинация метрик: размер модели в памяти (Flash), потребление оперативной памяти (RAM) во время вывода, скорость вывода (латентность), энергопотребление (если доступны модели) и отклонение точности от целевого значения. Для эффективного исследования пространства конфигураций используются методы байесовской оптимизации или эволюционные алгоритмы [15]. Механизм выполняет итеративный процесс: выбор конфигурации -> применение конвейера с данной конфигурацией -> оценка результирующей модели на эмуляторе целевого МК или с использованием точных моделей ресурсов -> обновление стратегии поиска на основе полученных метрик. Критерием остановки является достижение целевой точности при минимальных

ресурсных затратах или исчерпание вычислительного бюджета оптимизации.

Заключение

Настоящая работа была посвящена решению актуальной проблемы развертывания моделей глубокого обучения на ресурсо-ограниченных микроконтроллерах семейства Cortex-M, что критически важно для развития Интернета вещей и периферийных вычислений. Прямое применение стандартных моделей на таких устройствах невозможно из-за жестких ограничений памяти, вычислительной мощности и энергопотребления. Хотя методы сжатия, такие как квантование, прунинг и дистилляция знаний, широко исследуются, их изолированное применение или использование в фиксированных комбинациях часто не позволяет достичь необходимого баланса между степенью сжатия, скоростью вывода и сохранением требуемой точности, особенно с учетом крайней гетерогенности платформ Cortex-M.

Для преодоления этих ограничений был предложен инновационный адаптивный конвейер сжатия. Его ключевая идея заключается в синергетическом последовательном применении структурированного прунинга, посттренировочного квантования (PTQ) и дистилляции знаний. Структурированный прунинг, удаляя избыточные каналы или фильтры, формирует оптимизированную архитектуру модели, что упрощает ее последующее выполнение на целевом МК. Применяемое затем квантование существенно снижает битность представления весов и активаций, уменьшая требования к памяти и вычислениям. Принципиально важно, что дистилляция знаний использует уже сжатую модель (после прунинга и квантования) в качестве "учителя". Это позволяет эффективно восстановить точность, потенциально утерянную на предыдущих агрессивных этапах сжатия, путем обучения компактной модели-ученика имитировать специфические знания, заложенные в

оптимизированной архитектуре, и адаптировать их под целевую платформу.

Сердцем предложенного подхода является адаптивный механизм автоматической конфигурации конвейера. Этот механизм, реализованный на основе методов оптимизации "черного ящика" (таких как байесовская оптимизация или эволюционные алгоритмы), автоматически подбирает оптимальные параметры каждого этапа (степень прунинга, битность квантования, параметры дистилляции, архитектуру ученика) и даже исследует порядок их применения. Целью оптимизации является достижение заданного уровня точности при минимизации ресурсных затрат – размера модели в ПЗУ (Flash), потребления ОЗУ (RAM), латентности вывода и энергопотребления – строго в соответствии со спецификацией конкретного микроконтроллера Cortex-M. Таким образом, данный подход предоставляет систематизированное и автоматизированное решение для эффективного развертывания сложных моделей ИИ на широком спектре крайне ограниченных периферийных устройств.

СПИСОК ЛИТЕРАТУРЫ:

1. Уорден П., Ситунаяке Д. TinyML: Машинное обучение с TensorFlow Lite на Arduino и сверхмаломощных микроконтроллерах / П. Уорден, Д. Ситунаяке; пер. с англ. – Москва: O'Reilly Media, 2019. – 504 с.
2. Han S. [и др.] Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding / S. Han, H. Mao, W.J. Dally // International Conference on Learning Representations (ICLR 2016). – 2016.
3. Gholami A. [и др.] A Survey of Quantization Methods for Efficient Neural Network Inference / A. Gholami, S. Kim, Z. Dong, Z. Yao, M.W. Mahoney, K. Keutzer // arXiv:2103.13630 [cs.LG]. – 2021.

4. Blalock D. [и др.] What is the State of Neural Network Pruning? / D. Blalock, J.J. Gonzalez Ortiz, J. Frankle, J. Guttag // Proceedings of Machine Learning and Systems (MLSys 2020). – 2020. – P. 129–146.
5. Lai L. Enabling Deep Learning at the IoT Edge / L. Lai, N. Suda // Proceedings of the International Conference on Computer-Aided Design (ICCAD 2018). – 2018. – Art. 113. – DOI 10.1145/3240765.3243488.
6. ARM Limited. Cortex-M Processors [Электронный ресурс] / ARM Limited. – URL: https://developer.arm.com/Processors/Cortex-M
7. Choudhary T. [и др.] Comprehensive survey on model compression and acceleration for deep neural networks / T. Choudhary, V. Mishra, A. Goswami, J. Sarangapani // Journal of Systems Architecture. – 2022. – Vol. 127. – Art. 102509. – DOI 10.1016/j.sysarc.2022.102509.
8. Krishnamoorthi R. Quantizing deep convolutional networks for efficient inference: A whitepaper / R. Krishnamoorthi // arXiv:1806.08342 [cs.CV]. – 2018. – Электрон. текстовые дан.
9. David R. [и др.] TensorFlow Lite Micro: Embedded Machine Learning on TinyML Systems / R. David, J. Duke, A. Jain, V. Janapa Reddi, N. Jeffries, J.-D. Li, N. Kreeger, I. Nappier, M. Natraj, S. Wang, P. Warden, R. Rhodes // arXiv:2010.08678 [cs.LG]. – 2020. – Электрон. текстовые дан.
10. STMicroelectronics. STM32Cube.AI [Электронный ресурс] / STMicroelectronics. – URL: https://www.st.com/en/embedded-software/x-cube-ai.html (дата обращения: 01.08.2025).
11. Lin J. [и др.] MCUNet: Tiny Deep Learning on IoT Devices / J. Lin, W.-M. Chen, Y. Lin, J. Cohn, C. Gan, S. Han // Advances in Neural Information Processing Systems (NeurIPS 2020). – 2020. – Vol. 33. – P. 11711–11722.

12. Lai L. [и др.] CMSIS-NN: Efficient Neural Network Kernels for Arm Cortex-M CPUs / L. Lai, N. Suda, V. Chandra // arXiv:1801.06601 [cs.LG]. – 2018. – Электрон. текстовые дан.
13. Nagel M. [и др.] A White Paper on Neural Network Quantization / M. Nagel, R.A. Amjad, M. van Baalen, C. Louizos, T. Blankevoort // arXiv:2106.08295 [cs.LG]. – 2021. – Электрон. текстовые дан.
14. Gou J. [и др.] Knowledge Distillation: A Survey / J. Gou, B. Yu, S.J. Maybank, D. Tao // International Journal of Computer Vision. – 2021. – Vol. 129. – P. 1789–1819. – DOI 10.1007/s11263-021-01453-z.
15. Falkner S. [и др.] BOHB: Robust and Efficient Hyperparameter Optimization at Scale / S. Falkner, A. Klein, F. Hutter // Proceedings of the 35th International Conference on Machine Learning (ICML 2018). – 2018. – P. 1436–1445.