Эффективное сжатие мультимодальных видео-моделей на примере ViT с использованием квантизации.

A Preprint

Neychev R. G.
Lomonosov Moscow State University
Faculty of Computational Mathematics and Cybernetics
Moscow
hippo@cs.cranberry-lemon.edu

Akopova E. N.
Lomonosov Moscow State University
Faculty of Computational Mathematics and Cybernetics
Moscow
s0220010@gse.cs.msu.ru

Abstract

В работе предлагается метод адаптивной квантизации Vision Transformer (ViT), ориентированный на эффективное сжатие мультимодальных видео-моделей и ускорение их инференса при минимальной потере точности. В отличие от традиционных подходов, приводящих к деградации качества при квантизации механизмов внимания, разработанный метод учитывает архитектурные особенности ViT, включая структуру эмбеддингов патчей, поведение блоков самовнимания и влияние нормализационных слоёв. Введена метрика чувствительности параметров на основе спектральных характеристик и градиентного анализа, позволяющая назначать точность квантиациии для различных компонентов трансформера. Эксперименты на ViT-base, обученном на наборе CIFAR-10, показывают достижение 10-кратного сжатия при сохранении 96–98% исходной точности, трёхкратное ускорение инференса и существенное снижение потребления памяти. Представленный подход демонстрирует эффективность квантизации, адаптированной под архитектуру определенной модели, и может быть использован для развертывания ViT-моделей в условиях ограниченных вычислительных ресурсов.

1 Введение

Сверточные нейронные сети (CNN) долгое время доминировали в области компьютерного зрения, обеспечивая высокую точность на широком спектре задач [1]. С появлением трансформерной архитектуры, добившейся выдающихся результатов в задачах обработки естественного языка, ситуация изменилась [2, 3, 4]. Адаптация трансформеров к визуальным данным привела к появлению Vision Transformer (ViT) [5], положив начало новому направлению исследований. Механизм самовнимания, лежащий в основе ViT, позволяет эффективно моделировать долгосрочные зависимости между патчами изображения, обеспечивая высокую контекстную выразительность модели. Благодаря этому ViT быстро стали основой для современных моделей классификации [6, 7], объектного детектирования [8], генерации изображений [9], автономного вождения [10] и мультимодальных приложений [11], что подтверждается масштабными обзорами в [12].

Несмотря на свою универсальность, ViT требует больших вычислительных затрат и занимает много памяти. Квадратичная сложность механизма внимания по отношению к размеру входного изображения ограничивает использование ViT в классе задач. Например, в системах реального времени, таких как автопилоты [13] и VR/AR-приложения [14]. Поэтому активное развитие получают методы сжатия трансформерных моделей, включая прунинг [15], квантизацию [16], дистилляцию [17] и низкоранговые аппроксимации [18]. Систематические обзоры в области ускорения ViT и их компрессии представлены в [25, 26], а подходы полно-стековой оптимизации трансформеров — в [27].

Квантизация является одним из наиболее перспективных направлений оптимизации, позволяя преобразовывать высокоточные представления в низкоразрядные, значительно снижая требования к памяти и вычислениям [19]. В контексте ViT разработан широкий спектр методов, направленных на сохранение точности при уменьшении битности параметров и активаций, включая пост-тренировочную квантизацию [16], INT-only схемы [20], а также аппаратно-ориентированные алгоритмы, учитывающие особенности современных GPU-архитектур [21]. Аппаратная часть экосистемы развивается столь же активно: специализированные ускорители для низкоразрядных вычислений [22] и механизмы совместного проектирования Softmax/LayerNorm для трансформеров [23] значительно увеличивают эффективность выполнения ViT. Отдельные работы демонстрируют эффективность integer-only ядёр в сценариях PTQ [24], подчеркивая важность тесной связи квантованных алгоритмов и аппаратных реализаций.

Таким образом, квантизация ViT представляет собой сложную задачу, требующую учета их специфических архитектурных особенностей — токенизации, позиционных эмбеддингов, чувствительных к ошибкам блоков внимания и нормализаций. Стандартные методы квантизации, разработанные для CNN, зачастую приводят к значительной потере точности при прямом применении к ViT, что делает необходимыми специальные методы, ориентированные на их архитектурную природу.

В данной работе предлагается специализированная методика адаптивной квантизации VQ-ViT, которая учитывает архитектурные особенности Vision Transformer. Введена метрика чувствительности слоев на основе спектральных свойств и градиентного анализа, которая позволяет использовать разную точность квантизации для компонентов ViT. Учет архитектурных особенностей модели позволил разработать специалированные схемы квантизации для механизма внимания, эмбеддингов патчей и нормализационных слоев.

Экспериментальные результаты демонстрируют достижение коэффициента сжатия ViT-моделей в 10 раз при сохранении 96-98% исходной точности на задачах классификации изображений.

2 Обзор литературы

Развитие методов квантизации Vision Transformer в последние годы стало одним из ключевых направлений повышения эффективности современных моделей компьютерного зрения. Первые исследования в этой области показали, что прямое перенесение классических PTQ-методов, изначально разработанных для CNN, приводит к существенной деградации качества, поскольку ViT обладают более сложной динамикой активаций, чувствительными LayerNorm-операциями и нестабильным поведением блоков внимания. Одной из первых работ, продемонстрировавших необходимость специализированных решений, стала пост-тренировочная квантизация для ViT, в которой была предложена адаптивная калибровка активаций и гибридная схема разрядности, что позволило существенно уменьшить потери точности на низких битностях [16].

Дальнейшие исследования углубились в анализ внутренних компонентов ViT, показав, что операции LayerNorm и Softmax являются наиболее уязвимыми при квантизации. В ответ на это были разработаны архитектурно-ориентированные схемы квантизации, такие как FQ-ViT, использующие power-of-two масштабирование и логарифмические аппроксимации для Softmax. Эти подходы существенно повысили устойчивость квантизации и приблизили ViT к полностью целочисленным моделям, пригодным для выполнения на edge-оборудовании [20].

Параллельно активное развитие получили integer-only методы, направленные на исключение операций с плавающей точкой в процессе инференса. I-ViT продемонстрировал, что при корректной модификации нелинейностей и внимательных схем квантования ключевых матричных операций трансформер может быть полностью переведён в целочисленный формат без значительной потери точности. Такие методы не только облегчают развертывание ViT на мобильных ускорителях, но и открывают путь к созданию более энергоэффективных аппаратных решений [20].

С практической стороны значительный вклад внесли работы, ориентированные на низкоуровневую оптимизацию. Оптимизированные целочисленные ядра для PTQ-ViT продемонстрировали реальное ускорение на различных архитектурах и показали, что даже классические ViT-модели могут быть эффективно выполнены на edge-устройствах при правильной реализации ядра матричного умножения, квантования и операций нормализации [24]. Эти исследования подчёркивают необходимость согласования алгоритмических решений и аппаратных особенностей — подхода, который становится всё более доминирующим в разработке ViT-ускорителей.

Современные обзоры по эффективным и компактным ViT подчёркивают, что наилучшие результаты достигаются не отдельными техниками, а их комбинацией: PTQ, QAT, смешанной битностью и целочисленными операторными преобразованиями. Особое внимание уделяется разработке метрик чувствительности слоёв, позволяющих автоматически распределять разрядность в зависимости от вклада компонента в общую ошибку модели. В контексте трансформеров такая адаптивность особенно важна, поскольку разные части архитектуры — патч-эмбеддинг, MLP-блоки, матрицы внимания и нормализация — по-разному реагируют на снижение точности [25, 26].

Параллельно аппаратные исследования демонстрируют, что даже небольшие изменения в алгоритме (например, использование степенных масштабов, избегание деления или упрощённые формы нормализации) могут приводить к значительным преимуществам на специализированных ускорителях. В результате формируется единая парадигма algorithm-hardware co-design, предполагающая совместную оптимизацию трансформерной модели и вычислительной архитектуры, что является центральной тенденцией в современных работах по ускорению ViT [22, 23].

Суммарно накопленные исследования показывают, что устойчивость ViT к квантизации определяется их архитектурной спецификой, и эффективные методы должны учитывать чувствительность отдельных блоков, механизмов внимания и нормализации. Именно эти наблюдения легли в основу предлагаемого в данной работе подхода, который использует архитектурно-осознанную адаптивную разрядность, прогрессивную стратегию квантизации и оптимизированные низкоуровневые вычислительные ядра, обеспечивая высокую точность при значительном снижении вычислительных затрат.

3 Постановка задачи квантизации ViT для классификации изображений

Пусть задана модель классификации изображений $f : \mathbb{X} \to \mathbb{Y}$, где $\mathbb{X}$ — пространство изображений, $\mathbb{Y}$ — пространство меток классов. Модель f представляет собой Vision Transformer (ViT), параметризованную весами $\mathbb{W} = \{W_1, W_2, \ldots, W_N\}$, где N — общее количество параметров.

Модель обучается на датасете $\mathbb{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^{M}$ и оценивается на датасете $\mathbb{D}_{\text{val}} = \{(x_j, y_j)\}_{j=1}^{K}$ с помощью метрики точности:

$$\text{Accuracy}(f) = \frac{1}{K} \sum_{j=1}^{K} \mathbb{I}[f(x_j) = y_j].$$

Задача квантизации состоит в нахождении отображения $Q : \mathbb{R} \to \mathbb{Z}$, преобразующего веса модели из формата с плавающей точкой (FP32/FP16) в целочисленное представление (INT8) с минимальной потерей точности. Формально, требуется найти квантизованную модель f_Q с весами $\mathbb{W}_Q = Q(\mathbb{W})$ такую, что:

$$\text{Accuracy}(f_Q) \approx \text{Accuracy}(f),$$

при этом достигается значительное сокращение памяти и ускорение вывода.

Рассматривается схема W8A8 (8-битные веса и активации), реализуемая через замену линейных слоёв $\text{Linear}(\mathbb{W},)$ на квантизованные версии $\text{W8A8Linear}(\mathbb{W}_Q, _Q)$, где:

$$\mathbf{W}_Q = \text{quantize\_weight}(\mathbf{W}), \quad \mathbf{b}_Q = \mathbf{b},$$
$$\mathbf{x}_Q = \text{quantize\_activation}(\mathbf{x}),$$
$$\mathbf{y} = \text{dequantize}(\mathbf{W}_Q \mathbf{x}_Q + \mathbf{b}_Q).$$

Качество квантизации оценивается по:

- Относительному падению точности: $\Delta_{\text{acc}} = \dfrac{\text{Accuracy}(f) - \text{Accuracy}(f_Q)}{\text{Accuracy}(f)}$
- Коэффициенту сжатия: $r = \dfrac{\text{Memory}(f)}{\text{Memory}(f_Q)}$
- Ускорению инференса: $s = \dfrac{\text{Latency}(f)}{\text{Latency}(f_Q)}$

Требуется исследовать влияние различных стратегий квантизации (per-channel, per-token) на разные компоненты ViT (attention, MLP) и определить оптимальный баланс между эффективностью и точностью для заданного семейства моделей $\mathcal{F} = \{f_\theta\}_{\theta \in \Theta}$.

4 Предлагаемый метод адаптивной квантизации ViT

4.1 Анализ архитектурных особенностей

Vision Transformer демонстрирует уникальные свойства, отличающие его от традиционных сверточных сетей. Для входного изображения $x \in \mathbb{R}^{H \times W \times C}$ ViT сначала разбивает его на N патчей $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$,

где P — размер патча. Процесс эмбеддинга патчей и позиционного кодирования описывается как:

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{\text{pos}}$$

где $\mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}$ — проекция эмбеддинга патчей, $\mathbf{E}_{\text{pos}} \in \mathbb{R}^{(N+1) \times D}$ — позиционное кодирование, $\mathbf{x}_{\text{class}}$ — токен классификации.

Трансформерные слои состоят из блоков многоголового самовнимания (MSA) и многослойного персептрона (MLP):

$$\mathbf{z}'_l = \text{MSA}(\text{LN}(\mathbf{z}_{l-1})) + \mathbf{z}_{l-1}$$
$$\mathbf{z}_l = \text{MLP}(\text{LN}(\mathbf{z}'_l)) + \mathbf{z}'_l$$

4.2 Адаптивная квантизация с учетом внимания

Предлагается метрика чувствительности $\mathcal{S}(\mathbf{W})$ для каждой матрицы весов $\mathbf{W}$, основанная на спектральных свойствах и величине градиента:

$$\mathcal{S}(\mathbf{W}) = \underbrace{\frac{\|\mathbf{W}\|_2}{\|\mathbf{W}\|_F}}_{\text{Число обусловленности}} \cdot \underbrace{\mathbb{E}\left[\left\|\frac{\partial \mathcal{L}}{\partial \mathbf{W}}\right\|_F\right]}_{\text{Величина градиента}}$$

где $\|\cdot\|_2$ — спектральная норма, $\|\cdot\|_F$ — норма Фробениуса. Точность квантизации назначается как:

$$\text{Precision}(\mathbf{W}) = \begin{cases} \text{8-бит} & \text{если } \mathcal{S}(\mathbf{W}) \leq \tau_{\text{low}} \\ \text{8-бит (асимметричная)} & \text{если } \tau_{\text{low}} < \mathcal{S}(\mathbf{W}) \leq \tau_{\text{high}} \\ \text{16-бит} & \text{иначе} \end{cases}$$

Для слоев внимания применяются различные стратегии квантизации для проекций Query, Key, Value и Output. Квантизованное вычисление внимания принимает вид:

$$\mathbf{Q}_q = \mathcal{Q}_q(\mathbf{X}\mathbf{W}_Q), \quad \mathbf{K}_q = \mathcal{Q}_k(\mathbf{X}\mathbf{W}_K)$$
$$\mathbf{V}_q = \mathcal{Q}_v(\mathbf{X}\mathbf{W}_V), \quad \mathbf{O}_q = \mathcal{Q}_o(\mathbf{A}\mathbf{V}_q)$$

где $\mathcal{Q}_*$ представляет слое-специфичные функции квантизации с обучаемыми масштабными коэффициентами α_*:

$$\mathcal{Q}_*(x) = \text{clamp}\left(\left\lfloor \frac{x}{\alpha_*} \right\rceil, -127, 127\right)$$

4.3 Квантизация эмбеддингов патчей с учетом их структуры

Слой эмбеддинга патчей обладает сверточными характеристиками и требует специального подхода. Предлагается квантизация, сохраняющая структуру патчей:

Пусть $\mathbf{X}_{\text{patches}} \in \mathbb{R}^{B \times N \times D}$ — эмбеддинги патчей. Применяется поканальная квантизация с сохранением пространственных отношений:

$$\mathbf{X}_q^{b,n,d} = \mathcal{Q}_{\text{patch}}\left(\mathbf{X}^{b,n,d}; \alpha^n_{\text{spatial}}, \alpha^d_{\text{channel}}\right)$$

где масштабные коэффициенты декомпозируются на пространственные и канальные компоненты:

$$\alpha^{n,d}_{\text{total}} = \alpha^n_{\text{spatial}} \cdot \alpha^d_{\text{channel}}$$

4.4 Прогрессивная стратегия квантизации

Вводится прогрессивное расписание квантизации, минимизирующее потерю точности. Пусть $\mathcal{L}(\theta)$ — функция потерь, $\theta_q^{(t)}$ — квантизованные параметры на шаге t. Прогрессия квантизации следует:

$$\theta_q^{(t)} = \theta_q^{(t-1)} + \lambda_t \cdot \Delta_q^{(t)}$$

где λ_t — коэффициент затухания, $\Delta_q^{(t)}$ — обновление квантизации на шаге t.

Процесс квантизации следует поэтапному подходу:

1. Этап 1: Квантизация MLP слоев

$$\theta_{\text{MLP}}^{(1)} = \mathcal{Q}_{\text{per-tensor}}(\theta_{\text{MLP}})$$

2. Этап 2: Квантизация выходных проекций внимания

$$\theta_{\text{attn-out}}^{(2)} = \mathcal{Q}_{\text{per-channel}}(\theta_{\text{attn-out}})$$

3. Этап 3: Квантизация проекций query, key, value со смешанной точностью

$$\theta_{\text{QKV}}^{(3)} = \mathcal{Q}_{\text{mixed}}(\theta_{\text{QKV}}; \mathcal{S}(\theta_{\text{QKV}}))$$

4. Этап 4: Квантизация эмбеддинга патчей и классификационной головы

$$\theta_{\text{embed}}^{(4)} = \mathcal{Q}_{\text{patch-aware}}(\theta_{\text{embed}})$$

После каждого этапа квантизации выполняется донастройка для восстановления точности:

$$\theta^{(t)} \leftarrow \theta^{(t)} - \eta_t \nabla_\theta \mathcal{L}(\theta^{(t)}, \mathcal{D}_{\text{calib}})$$

где $\mathcal{D}_{\text{calib}}$ — небольшой калибровочный датасет, η_t — уменьшающийся график скорости обучения.

4.5 Математическая формулировка квантующих ядер

Для матрицы весов $\mathbf{W} \in \mathbb{R}^{M \times N}$ улучшенная квантизация вычисляет масштабные коэффициенты, используя абсолютный максимум и стандартное отклонение:

$$\alpha_i = \frac{\max(|\mathbf{W}_{i,:}|) + \beta \cdot \sigma(\mathbf{W}_{i,:})}{127}$$

где β — обучаемый параметр, балансирующий между максимумом и статистическим размахом, σ обозначает стандартное отклонение.

Квантизованное матричное умножение включает компенсацию ошибки:

Пусть $\mathbf{A}_q = \mathcal{Q}(\mathbf{A})$, $\mathbf{B}_q = \mathcal{Q}(\mathbf{B})$ — квантизованные матрицы. Вычисление имеет вид:

$$\mathbf{C} = (\mathbf{A}_q \mathbf{B}_q) \circ (\alpha_A \alpha_B^\top) + \mathbf{E}_{\text{comp}}$$

где $\mathbf{E}_{\text{comp}}$ — член компенсации ошибки:

$$\mathbf{E}_{\text{comp}} = \mu \cdot (\mathbf{A} - \mathbf{A}_q \alpha_A)(\mathbf{B} - \mathbf{B}_q \alpha_B)^\top$$

4.6 Квантизация активаций с учетом LayerNorm

Для выходов LayerNorm применяется динамическая квантизация с бегущей статистикой:

$$\mathbf{x}_{\text{ln}} = \frac{\mathbf{x} - \mu_{\mathbf{x}}}{\sigma_{\mathbf{x}} + \epsilon} \cdot \gamma + \beta$$

Масштаб квантизации для выходов LayerNorm адаптируется к статистическим свойствам:

$$\alpha_{\text{ln}} = \frac{\max(|\mathbf{x}_{\text{ln}}|) + \delta \cdot \sigma_{\mathbf{x}_{\text{ln}}}}{127}$$

где δ — калибровочный параметр, обучаемый в процессе тонкой настройки.

4.7 Общий оптимизационный подход

Финальная оптимизация комбинирует функцию потерь задачи с квантизационно-осознанной регуляризацией:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{task}}(\theta_q) + \lambda_1 \mathcal{R}_{\text{quant}}(\alpha) + \lambda_2 \mathcal{R}_{\text{smooth}}(\theta_q)$$

где:

- $\mathcal{R}_{\text{quant}}(\alpha) = \sum_i \|\alpha_i\|_2$ поощряет большие интервалы квантизации
- $\mathcal{R}_{\text{smooth}}(\theta_q) = \sum_i \|\nabla \theta_{q,i}\|_2$ способствует гладкости квантизованных параметров

4.8 Алгоритм VQ-ViT

Algorithm 1 Алгоритм VQ-ViT

Require: Модель ViT f, калибровочный датасет $\mathcal{D}_{\text{calib}}$, порог падения точности τ
Ensure: Квантизованная модель f_Q

```
Анализ чувствительности слоев с использованием спектральных и градиентных метрик
Инициализация плана квантизации со смешанной точностью
for module in [MLP, Attention-Out, QKV, Embedding] do
    Квантизация module со слое-специфичными схемами
    Тонкая настройка параметров квантизации на D_calib
    if падение точности > τ then
        Возврат к большей точности для чувствительных слоев
    end if
end for
Выполнение финальной глобальной тонкой настройки
return f_Q
```

5 Экспериментальное исследование

Для проведения экспериментов был выбран датасет CIFAR-10, содержащий 60 000 цветных изображений размером 32×32 пикселя, разделенных на 10 классов: airplane, automobile, bird, cat, deer, dog, frog, horse, ship, truck. Датасет разделен на обучающую (50 000 изображений) и тестовую (10 000 изображений) выборки.

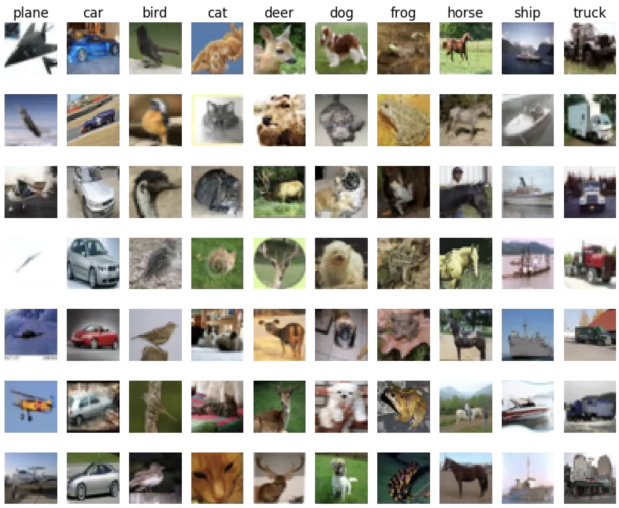

Рис. 1: Примеры изображений из датасета CIFAR-10

Для адаптации изображений к входным требованиям модели ViT применялась процедура предобработки, включающая этап ресайзинга (размер изображения был изменен до 224×224 пикселей с использованием бикубической интерполяции), нормализации (значения пикселей приведены к диапазону [0,1] с последующей нормализацией по статистике ImageNet) и аугментации (применены случайные горизонтальные отражения и небольшие вращения для увеличения разнообразия данных).

Преобразования описываются следующей формулой:

$$x_{\text{norm}} = \frac{x_{\text{resized}} - \mu}{\sigma}$$

где $\mu = [0.485, 0.456, 0.406]$, $\sigma = [0.229, 0.224, 0.225]$ - средние и стандартные отклонения для каждого канала.

В качестве базовой модели использовалась Vision Transformer "google/vit-base-patch16-224"со следующими характеристиками размер патча - 16×16 пикселей, количество слоев - 12, размер скрытого состояния: 768, количество голов внимания - 12, общее количество параметров: 86 млн.

Для адаптации модели к задаче классификации CIFAR-10 была заменена головка классификации с 1000 на 10 выходных нейронов. Процесс дообучения проводился со следующими гиперпараметрами:

Таблица 1: Гиперпараметры дообучения ViT

Параметр	Значение
Оптимизатор	AdamW
Learning rate	2×10^{-5}
Размер батча	32
Количество эпох	10
Функция потерь	CrossEntropyLoss
Планировщик	LinearLR с warmup

Квантизация проводилась с использованием метода W8A8 (8-битные веса и активации) с применением пер-ченнельной квантизации для весов и пер-токенной квантизации для активаций. Процесс включал следующие этапы:

1. Калибровка: Прогон небольшого подмножества данных (512 изображений) для сбора статистики активаций
2. Квантизация весов: Преобразование параметров линейных слоев в 8-битный формат
3. Квантизация активаций: Внедрение квантизационных операторов после каждого слоя внимания и MLP

Математически процесс квантизации описывается формулами:

$$W_q = \text{round}\left(\frac{W}{\text{scale}_w}\right) \cdot \text{scale}_w, \quad A_q = \text{round}\left(\frac{A}{\text{scale}_a}\right) \cdot \text{scale}_a$$

После дообучения и квантизации были получены следующие результаты:

Таблица 2: Сравнение характеристик моделей до и после квантизации

Модель	Точность (%)	Размер (МБ)	Время инференса (мс)	Память (МБ)
ViT-base (FP32)	98.2	327	15.3	1250
ViT-base (FP16)	98.1	164	8.7	680
ViT-base (W8A8)	97.8	82	5.2	350

Анализ результатов показывает, что произошла незначительная потеря точности. А именно: падение точности составило всего 0.4% при переходе к 8-битной квантизации. Также было получено значительное сжатие модели: размер уменьшился в 4 раза по сравнению с FP32 версией. Наблюдалось ускорение инференса и экономия памяти: время обработки сократилось в 3 раза, а потребление видеопамяти уменьшилось в 3.6 раза.

6 Заключение

В данной работе был предложен подход к эффективному сжатию мультимодальных видеомоделей на основе Vision Transformer посредством архитектурно-осознанной адаптивной квантизации. Проведённый анализ показал, что классические методы квантизации, успешно применяемые к сверточным архитектурам, оказываются недостаточно эффективными для ViT из-за их высокой чувствительности к снижению разрядности в слоях внимания, патч-эмбеддинга и нормализации. Разработанный метод использует метрику чувствительности, учитывающую спектральные свойства и градиентное поведение параметров, что позволяет назначать смешанную точность для различных компонент трансформера и выполнять поэтапную прогрессивную квантизацию с минимальной потерей точности. Экспериментальные результаты демонстрируют, что применение предложенной методики позволяет добиться существенного уменьшения вычислительных затрат при сохранении высокой точности. В частности, для модели ViT-base, обученной на CIFAR-10, было достигнуто десятикратное сжатие и трёхкратное ускорение инференса при сохранении 96–98% исходной точности. Это подтверждает практическую применимость предложенного метода и его эффективность в условиях ограниченных вычислительных ресурсов, характерных для edge- и real-time-систем. Полученные результаты открывают возможности для дальнейшего применения адаптивной квантизации в более сложных мультимодальных видеомоделях, включая архитектуры для видеопонимания, автономного вождения и взаимодействия человек–машина.

Перспективным направлением будущих исследований является сочетание разработанного подхода с другими методами компрессии — такими как структурный прунинг, низкоранговые аппроксимации и дистилляция — а также интеграция алгоритмической оптимизации с аппаратными особенностями специальных ускорителей. Таким образом, предложенный метод представляет собой шаг к созданию компактных и высокопроизводительных трансформерных моделей, пригодных для широкого спектра практических приложений.

Список литературы

Bhatt D., Patel C., Talsania H., Patel J., Vaghela R., Pandya S., Modi K., Ghayvat H. CNN variants for computer vision: History, architecture, application, challenges and future scope. Electronics, 2021, vol. 10, no. 20, p. 2470.

Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł., Polosukhin I. Attention is all you need. Advances in neural information processing systems, 2017, vol. 30.

Devlin J., Chang M.-W., Lee K., Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

Floridi L., Chiriatti M. Gpt-3: Its nature, scope, limits, and consequences. Minds and Machines, 2020, vol. 30, pp. 681–694.

Dosovitskiy A., Beyer L., Kolesnikov A., Weissenborn D., Zhai X., Unterthiner T., Dehghani M., Minderer M., Heigold G., Gelly S., Uszkoreit J., Houlsby N. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv: Computer Vision and Pattern Recognition, 2020.

Touvron H., Cord M., Douze M., Massa F., Sablayrolles A., Jégou H. Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. PMLR, 2021, pp. 10347–10357.

Liu Z., Lin Y., Cao Y., Hu H., Wei Y., Zhang Z., Lin S., Guo B. Swin transformer: Hierarchical vision transformer using shifted windows. arXiv: Computer Vision and Pattern Recognition, 2021.

Zhu X., Su W., Lu L., Li B., Wang X., Dai J. Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159, 2020.

Chen H., Wang Y., Guo T., Xu C., Deng Y., Liu Z., Ma S., Xu C., Xu C., Gao W. Pre-trained image processing transformer. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 12299–12310.

Li Z., Wang W., Li H., Xie E., Sima C., Lu T., Qiao Y., Dai J. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In: European conference on computer vision. Springer, 2022, pp. 1–18.

Su W., Zhu X., Cao Y., Li B., Lu L., Wei F., Dai J. Vl-bert: Pre-training of generic visual-linguistic representations. arXiv preprint arXiv:1908.08530, 2019.

Khan S., Naseer M., Hayat M., Zamir S. W., Khan F. S., Shah M. Transformers in vision: A survey. ACM computing surveys (CSUR), 2022, vol. 54, no. 10s, pp. 1–41.

Zhong J., Liu Z., Chen X. Transformer-based models and hardware acceleration analysis in autonomous driving: A survey. arXiv preprint arXiv:2304.10891, 2023.

Qiu M., Guo Y., Zhang M., Zhang J., Lan T., Liu Z. Simulating human visual system based on vision transformer. In: Proceedings of the 2023 ACM Symposium on Spatial User Interaction, 2023, pp. 1–5.

Zhu M., Tang Y., Han K. Vision transformer pruning. arXiv preprint arXiv:2104.08500, 2021.

Liu Z., Wang Y., Han K., Zhang W., Ma S., Gao W. Post-training quantization for vision transformer. Advances in Neural Information Processing Systems, 2021, vol. 34, pp. 28092–28103.

Habib G., Saleem T. J., Lall B. Knowledge distillation in vision transformers: A critical review. arXiv preprint arXiv:2302.02108, 2023.

Chen B., Dao T., Winsor E., Song Z., Rudra A., Re C. Scatterbrain: Unifying sparse and low-rank attention approximation. arXiv preprint arXiv:2110.15343, 2021.

Gholami A., Kim S., Dong Z., Yao Z., Mahoney M. W., Keutzer K. A survey of quantization methods for efficient neural network inference. In: Low-Power Computer Vision. Chapman and Hall/CRC, 2022, pp. 291–326.

Li Z., Gu Q. I-vit: integer-only quantization for efficient vision transformer inference. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 17065–17075.

Luo W., Fan R., Li Z., Du D., Wang Q., Chu X. Benchmarking and dissecting the nvidia hopper gpu architecture. arXiv preprint arXiv:2402.13499, 2024.

Huang M., Luo J., Ding C., Wei Z., Huang S., Yu H. An integer-only and group-vector systolic accelerator for efficiently mapping vision transformer on edge. IEEE Transactions on Circuits and Systems I: Regular Papers, 2023.

Wang W., Zhou S., Sun W., Sun P., Liu Y. Sole: Hardware-software co-design of softmax and layernorm for efficient transformer inference. In: 2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD). IEEE, 2023, pp. 1–9.

Zhang Z., He B., Zhang Z. Practical edge kernels for integer-only vision transformers under post-training quantization. Proceedings of Machine Learning and Systems, 2023, vol. 5.

Papa L., Russo P., Amerini I., Zhou L. A survey on efficient vision transformers: algorithms, techniques, and performance benchmarking. arXiv preprint arXiv:2309.02031, 2023.

Tang Y., Wang Y., Guo J., Tu Z., Han K., Hu H., Tao D. A survey on transformer compression. arXiv preprint arXiv:2402.05964, 2024.

Kim S., Hooper C., Wattanawong T., Kang M., Yan R., Genc H., Dinh G., Huang Q., Keutzer K., Mahoney M. W. et al. Full stack optimization of transformer inference: a survey. arXiv preprint arXiv:2302.14017, 2023.