Калифорнийские умы сотворили невозможное благодаря 3-битным весам.

Исследователи из Калифорнийского университета в Санта-Крузе, Сучжоуского университета и Калифорнийского университета в Дэвисе представили новую архитектуру для языковых моделей, которая устраняет необходимость в матричных умножениях (MatMul). Эта инновация позволяет значительно сократить затраты памяти и времени на обучение и работу моделей.

Матричное умножение (MatMul) является одной из самых ресурсоёмких операций в моделях-трансформерах. По мере увеличения размеров языковых моделей, растут и затраты на MatMul, что требует больших вычислительных ресурсов и неминуемо ведёт к задержкам.

В своей работе исследователи предложили MatMul-free модели, которые показывают производительность на уровне современных трансформеров, но требуют значительно меньше памяти для выполнения. В отличие от традиционных моделей, использующих 16-битные веса, новая архитектура применяет 3-битные веса, принимающие значения -1, 0 и +1, что сильно снижает вычислительные затраты.

Использование таких тернарных весов (состоящих из трёх компонентов) позволяет заменить MatMul на операции сложения и вычитания, что существенно упрощает вычисления. В новой архитектуре применены так называемые «BitLinear слои», которые используют тернарные веса для достижения схожих результатов при меньших затратах.

Исследователи также предложили замену традиционного токен-миксера на MatMul-free Linear Gated Recurrent Unit (MLGRU). Эта модель обновляет скрытые состояния с помощью простых тернарных операций, обходясь без дорогостоящих матричных умножений.

Вместо традиционного канального миксера используется модифицированный Gated Linear Unit (GLU), адаптированный для работы с тернарными весами. Это позволяет снизить вычислительную сложность и потребление памяти, сохраняя эффективность интеграции признаков.

Исследователи сравнили две вариации своей модели с архитектурой Transformer++ (используемой в Llama-2) и обнаружили, что их новые модели более эффективно используют дополнительные вычислительные ресурсы для улучшения производительности.

MatMul-free модели также продемонстрировали превосходство на ряде языковых задач. Например, модель с 2.7 миллиардами параметров превзошла Transformer++ на двух сложных тестах (ARC-Challenge и OpenbookQA), сохраняя сопоставимую производительность в других задачах.

Ожидаемо, MatMul-free модели обладают меньшим использованием памяти и задержками по сравнению с Transformer++. Для модели с 13 миллиардами параметров MatMul-free потребляла только 4.19 ГБ памяти и имела задержку 695.48 мс, тогда как Transformer++ требовала 48.50 ГБ памяти и имела задержку 3183.10 мс.

Исследователи также разработали оптимизированную Графический процессор, или GPU (Graphics Processing Unit), это вычислительное устройство, спроектированное специально для обработки графики и параллельных вычислений. Оно используется для ускорения операций, связанных с отображением изображений, видео и 3D-графикой на компьютере или другом устройстве.

 Благодаря своей способности эффективно выполнять множество однотипных операций одновременно, GPU также стали неотъемлемой частью для выполнения разнообразных вычислительных задач, таких как научные исследования, машинное обучение, криптовалютное майнинг и многое другое." data-html="true" data-original-title="GPU" >GPU-реализацию и специальную FPGA-конфигурацию для MatMul-free моделей. Это позволило ускорить обучение на 25.6% и сократить потребление памяти на 61% по сравнению с неоптимизированной реализацией.

Авторы работы полагают, что их исследования могут проложить путь к разработке более эффективных и дружественных к оборудованию архитектур глубокого обучения.

Из-за ограничений в вычислительных ресурсах, им не удалось протестировать архитектуру на моделях с более чем 100 миллиардами параметров, однако исследователи надеются, что их работа вдохновит другие учреждения на создание и использование подобных лёгких моделей.

В идеале такая архитектура сделает языковые модели гораздо менее зависимыми от высокопроизводительных графических процессоров, таких как Nvidia, и позволит исследователям запускать мощные модели на более бюджетных типах процессоров, которые, к тому же, будет куда проще достать в эпоху повсеместного машинного обучения.

Код алгоритма и всех моделей уже доступен для исследовательского сообщества, что позволит совместными усилиями и абсолютно прозрачно развивать и улучшать данную

архитектуру в будущем.