УДК 004.8

ОЦЕНКА ПРОИЗВОДИТЕЛЬНОСТИ РАЗЛИЧНЫХ ПОДХОДОВ ПАРАЛЛЕЛЬНОГО ЗАПУСКА МОДЕЛЕЙ BERT НА GPU

Бадретдинов Д.В., студент гр. ПМР-221, III курс
Научный руководитель: Зайченко Е.А., ст. преподаватель
Белорусско-Российский университет
г. Могилёв

В настоящее время технологии обработки естественного языка на основе глубоких нейронных сетей получили широкое распространение во многих областях, включая автоматизацию текстового анализа и построение интеллектуальных систем. Одними из наиболее востребованных моделей для подобных задач являются модели семейства BERT. Часто возникает необходимость одновременного выполнения множества таких моделей на графических ускорителях (GPU), однако такой подход требует эффективного управления ресурсами и грамотного выбора стратегии запуска моделей.

Существуют разные способы организации параллельного запуска моделей BERT: синхронный (последовательный), асинхронный и многопоточный. Каждый из этих подходов обладает своими особенностями с точки зрения производительности и затрат видеопамяти. В данной работе представлены результаты сравнительного анализа производительности указанных подходов при одновременном запуске моделей различных архитектур (ruRoberta и rembert) на нескольких типах современных GPU: GeForce RTX 4070, RTX 6000Ada, RTX 5000Ada и A100 SXM4.

Полученные в ходе исследования данные позволяют выявить наиболее эффективный подход, обеспечивающий минимальное время выполнения и низкие затраты памяти вне зависимости от архитектуры моделей и используемого аппаратного обеспечения.

Целью данного исследования является сравнительный анализ производительности синхронного, асинхронного и многопоточного подходов параллельного запуска моделей BERT различных архитектур на графических ускорителях (GPU), а также выявление оптимального подхода, обеспечивающего минимальное время выполнения и наименьшие затраты видеопамяти вне зависимости от используемой модели и видеокарты.

Для оценки эффективности различных подходов к параллельному запуску моделей BERT были проведены эксперименты с использованием двух архитектур моделей: ruRoberta и rembert [1]. В качестве аппаратной платформы были выбраны четыре типа видеокарт: GeForce RTX 4070, RTX 6000Ada, RTX 5000Ada и A100 SXM4. Сравнивались три подхода к запуску моделей:

1) Синхронный (последовательный запуск моделей одна за другой);

2) Асинхронный (параллельный запуск с помощью асинхронных вызовов на основе event loops);

3) Многопоточный (параллельный запуск моделей с использованием многопоточности).

В процессе экспериментов замерялось общее время выполнения инференса (процесса получения ответа модели на основе входных данных; включает вычисления, задержку и выдачу результата) моделей при варьировании количества одновременно запускаемых моделей от одной до нескольких [2].

Для проведения исследования были реализованы три отдельных скрипта на языке Python, которые отражают синхронный, асинхронный и многопоточный подходы к параллельному запуску моделей BERT. Во всех подходах использовалась одна и та же предварительно обученная модель, сохранённая в отдельном файле, а также одинаковый текст для токенизации и подачи на вход моделям. Инференс моделей выполнялся на видеокарте, при этом входные данные предварительно загружались в память GPU [3]. Время выполнения каждого подхода измерялось многократно, затем рассчитывалось среднее значение, исключив из расчёта первые два замера для стабилизации производительности и устранения эффекта разогрева GPU.

Синхронный подход заключался в последовательном запуске моделей. В рамках каждой итерации эксперимента модели обрабатывались одна за другой, без дополнительного распараллеливания или использования других механизмов управления потоками выполнения. Время замерялось от запуска первой модели до завершения работы последней в последовательности.

Асинхронный подход был реализован с использованием встроенных механизмов асинхронного программирования на основе библиотеки asyncio. Для каждой модели создавалась отдельная асинхронная задача, что позволяло запускать инференс параллельно. Фактическое выполнение инференса осуществлялось в отдельных исполнителях (executors), которые запускались параллельно и ожидались одновременно с помощью функции библиотеки asyncio. Время выполнения измерялось от запуска всех асинхронных задач до завершения их параллельного выполнения.

Многопоточный подход реализовывался путём создания отдельных потоков для каждой модели. Каждый поток запускал инференс независимо и параллельно остальным потокам. После запуска всех потоков производилось ожидание их завершения. Замер времени осуществлялся от момента начала ожидания завершения всех потоков до фактического окончания выполнения последнего потока.

Таким образом, реализованные подходы отличаются именно способами организации параллельного запуска моделей, что позволило на практике выявить наиболее эффективный способ обработки моделей BERT на видеокартах.

На рисунках 1-2 представлены результаты замеров времени выполнения каждого из подходов (синхронного, асинхронного и многопоточного) при разном количестве одновременно запущенных моделей BERT (rembert, ruRoberta) на видеокарте RTX 6000Ada. Данные замеры наглядно демонстрируют эффективность каждого подхода, а также позволяют оценить влияние архитектуры модели и типа используемого GPU на итоговую производительность.

Кол-во моделей	Синхронный	Асинхронный	Многопоточный
1.	0,01162	0,01238	0,01180
2.	0,02393	0,02290	0,02355
3.	0,03535	0,03144	0,03319
4.	0,04847	0,05776	0,04826
5.	0,06234	0,09879	0,06336
6.	0,07512	0,12965	0,07527
7.	0,08611	0,14471	0,08950
8.	0,10213	0,17297	0,10855
9.	0,10853	0,20210	0,12764
10.	0,12412	0,23320	0,15439
11.	0,13322	0,26363	0,17803
12.	0,15629	0,28720	0,19769
13.	0,16204	0,30790	0,22660
14.	0,17313	0,33227	0,26509
15.	0,18014	0,36067	0,31320
16.	0,19724	0,39550	0,36482
17.	0,20289	0,41876	0,39581
18.	0,22466	0,45015	0,42895
19.	0,23500	OutOfMemoryError	OutOfMemoryError

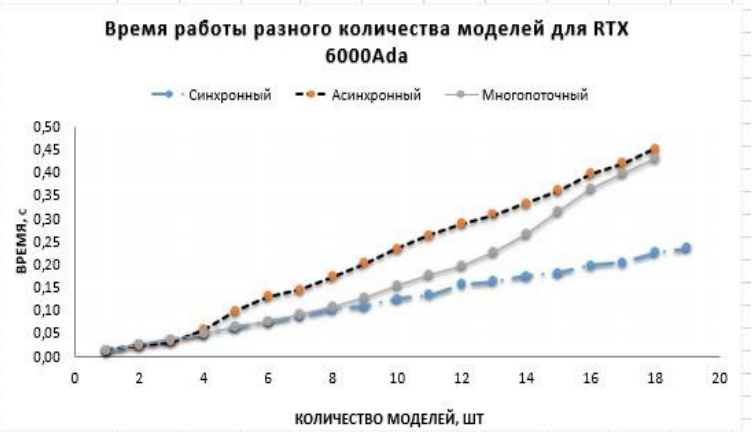

Рисунок 1 – Результаты экспериментов для RTX 6000Ada BERT: rembert

Рисунок 2 – Результаты экспериментов для RTX 6000Ada, BERT: ruRoberta

На основании проведенных экспериментов можно сделать следующие заключения:

1) Синхронный подход продемонстрировал наименьшее время выполнения во всех случаях и при любом количестве одновременно запускаемых моделей. Данный подход характеризовался минимальным ростом времени выполнения при увеличении числа параллельно запускаемых моделей, что связано с отсутствием дополнительных издержек на параллелизацию.

2) Асинхронный подход, несмотря на теоретическую возможность параллельной обработки и экономию времени ожидания, продемонстрировал более значительный рост времени выполнения с увеличением количества моделей. Это объясняется дополнительными накладными расходами на организацию асинхронного взаимодействия и очередей задач.

3) Многопоточный подход также уступал синхронному по скорости выполнения. С увеличением числа моделей существенно возрастала нагрузка на видеопамять и увеличивались затраты времени, связанные с управлением потоками и конкуренцией за ресурсы GPU.

Дополнительно стоит отметить, что характер описанных результатов не менялся при использовании различных архитектур моделей (ruRoberta или rembert) и разных моделей видеокарт (RTX 4070, RTX 6000Ada, RTX 5000Ada, A100 SXM4). Независимо от оборудования и архитектуры, синхронный подход оставался наиболее выгодным с точки зрения производительности и затрат видеопамяти, что подтверждается эмпирическими результатами.

По результатам проведённого исследования был осуществлён сравнительный анализ синхронного, асинхронного и многопоточного подходов параллельного запуска моделей BERT различных архитектур на графических ускорителях (GPU). Эмпирические результаты экспериментов подтвердили, что синхронный (последовательный) подход является наиболее эффективным и предпочтительным с точки зрения производительности и затрат видеопамяти. При увеличении количества одновременно работающих моделей именно синхронный подход демонстрировал наименьший рост времени выполнения и наиболее стабильные показатели.

Также было установлено, что указанные выводы не зависят от конкретной архитектуры модели (ruRoberta, rembert) и типа GPU (GeForce RTX 4070, RTX 6000Ada, RTX 5000Ada и A100 SXM4).

Таким образом, в практических задачах, требующих одновременного запуска множества моделей BERT, целесообразно использовать синхронный подход, что позволит значительно сократить затраты времени и ресурсов GPU. Представленные результаты могут быть полезны в прикладных и исследовательских проектах, связанных с массовой обработкой данных с использованием нейронных моделей.

Список литературы:

1. Муратулы, А. Искусственные нейронные сети. Нейросетевые технологии / А. Муратулы // Молодой ученый. – 2024. – № 25(524). – С. 97-99. – EDN DYFJIU.

2. Программное средство для построения поведенческой оценки человека на основе анализа тональностей текстов из социальных сетей / А. И. Мартышкин, Д. В. Пащенко, Р. А. Бикташев, А. А. Зоткина // Цифровая Индустрия: Состояние и Перспективы Развития 2023 (ЦИСП'2023) : Сборник научных статей, Челябинск, 21–23 ноября 2023 года. – Челябинск: Издательский центр Южно-Уральского государственного университета, 2024. – С. 353-363. – EDN MNLNVO.

3. Мелихова, Д. А. Искусственный интеллект для работы с данными / Д. А. Мелихова, О. А. Денисова // Актуальные проблемы технических, естественных и гуманитарных наук : Материалы Международной научно-технической конференции. Памяти В.Х. Хамаева, Уфа, 11–18 ноября 2024 года. – Уфа: УГНТУ, 2024. – С. 366-369. – EDN QWLYOY.