ORES
Научно-практический электронный журнал
Оригинальные исследования (ОРИС), *ISSN 2222-0402*
www.ores.su
original-research.ru

УДК 004.62

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ТЕХНОЛОГИЙ ОБРАБОТКИ БОЛЬШИХ CSV-ОТЧЁТОВ WILDBERRIES

Романовский Илья Олегович,
Студент группы ИУК5-21М
Калужский филиал Московского государственного технического университета имени Н.Э. Баумана
romanovskiyio@student.bmstu.ru

Антипова Ольга Викторовна,
Старший преподаватель кафедры ИУК5 КФ
Калужский филиал Московского государственного технического университета имени Н.Э. Баумана
antipovaov@bmstu.ru

Серпинский Роман Эдуардович,
Студент группы ИУК5-21М
Калужский филиал Московского государственного технического университета имени Н.Э. Баумана
serpinskiyrea@student.bmstu.ru

Аннотация

В данной работе проводится экспериментальное исследование технологий обработки больших CSV-отчётов маркетплейса Wildberries в задачах аналитической обработки данных. Рассматриваются три подхода: использование библиотеки pandas, библиотеки polars и локального аналитического движка ClickHouse. Актуальность исследования обусловлена тем, что отчёты маркетплейса могут содержать сотни тысяч и миллионы строк, а их обработка напрямую влияет на скорость построения аналитических витрин, расчёт рекламных и финансовых показателей, а также нагрузку на backend-сервис. Экспериментальная часть включает генерацию тестовых CSV-наборов объёмом 10, 50 и 100 МБ, выполнение единого аналитического сценария и сравнение технологий по времени обработки и пиковому потреблению оперативной памяти. На основе полученных результатов сформулированы выводы о применимости каждой технологии для задач аналитики Wildberries.

Ключевые слова: Wildberries, CSV-отчёты, pandas, polars, ClickHouse, аналитическая обработка данных, benchmark, производительность, использование памяти, агрегация данных, marketplace analytics.

COMPARATIVE ANALYSIS OF WILDBERRIES LARGE CSV REPORT PROCESSING TECHNOLOGIES

Ilya O. Romanovskiy,
Student of group IUK5-21M
Bauman Moscow State Technical University (Kaluga Branch)
romanovskiyio@student.bmstu.ru

Olga V. Antipova,
Senior Lecturer at the IUK5 CF Department
Kaluga Branch of the Bauman Moscow State Technical University
antipovaov@bmstu.ru

Roman E. Serpinski,
Student of the IUK5-21M group
Kaluga Branch of the Bauman Moscow State Technical University
serpinskiyrea@student.bmstu.ru

ABSTRACT

In this paper, we conduct an experimental study of technologies for processing large CSV reports from the Wildberries marketplace in analytical data processing tasks. Three approaches are being considered: using the pandas library, the polars library, and the ClickHouse local analysis engine. The relevance of the study is due to the fact that marketplace reports can contain hundreds of thousands and millions of lines, and their processing directly affects the speed of building analytical storefronts, calculating advertising and financial indicators, as well as the load on the backend service. The experimental part includes the generation of 10, 50, and 100 MB CSV test sets, the execution of a single analytical scenario, and a comparison of technologies in terms of processing time and peak RAM consumption. Based on the results obtained, conclusions are formulated about the applicability of each technology for Wildberries analytics tasks.

Keywords: Wildberries, CSV reports, pandas, polars, ClickHouse, analytical data processing, benchmark, performance, memory usage, data aggregation, marketplace analytics.

Введение

В аналитических сервисах, работающих с данными маркетплейсов, одним из ключевых этапов является обработка выгрузок в формате CSV. Такие выгрузки могут включать сведения о продажах, заказах, остатках, рекламных кампаниях, расходах и финансовых показателях. При малых объёмах данных обработка CSV обычно не вызывает существенных трудностей, однако при росте количества строк и ширины таблиц возникают проблемы с временем загрузки, использованием оперативной памяти и масштабируемостью вычислений.

Для сервисов аналитики Wildberries эти ограничения особенно важны, поскольку пользователь ожидает быстрое построение отчётов: выручки по дням, эффективности рекламных кампаний, прибыли по товарам, динамики заказов и показателей вроде ROAS, CTR и CPO. Если обработка CSV становится узким местом, это приводит к задержкам в интерфейсе, росту нагрузки на сервер и снижению устойчивости системы.

Целью работы является экспериментальное сравнение pandas, polars и ClickHouse при обработке CSV-отчётов Wildberries-подобной структуры. Основное внимание

уделяется практическим метрикам: общему времени обработки, времени загрузки CSV, времени вычислений после загрузки и пиковому потреблению оперативной памяти.

Используемые технологии и критерии сравнения

В качестве сравниваемых технологий были выбраны три инструмента, которые могут применяться при построении аналитического backend-сервиса [1].

pandas — классическая Python-библиотека для табличной обработки данных, широко применяемая в аналитике и прототипировании [2].

polars — современная DataFrame-библиотека, ориентированная на высокую скорость обработки и эффективное использование многопоточности [5].

ClickHouse local — локальный режим ClickHouse, позволяющий выполнять SQL-запросы непосредственно к CSV-файлам без развёртывания полноценного сервера [6].

Для сравнения применялись следующие метрики.

total_seconds — полное время выполнения аналитического сценария.

load_seconds — время загрузки CSV-файла в память; для ClickHouse не выделялось отдельно, так как чтение CSV выполнялось внутри SQL-запроса.

compute_seconds — время фильтрации, расчёта производных показателей и агрегации после загрузки данных.

peak_rss_mb — пиковое потребление оперативной памяти. Для pandas и polars измерялся текущий Python-процесс [4], для ClickHouse — дочерний процесс.

Практическая часть

В рамках практической части был реализован экспериментальный стенд, включающий генератор тестовых данных, отдельные benchmark-сценарии для каждой технологии и модуль формирования сводных результатов. Тестовые данные имитируют структуру отчётов Wildberries и содержат поля даты заказа, артикула, бренда, рекламной кампании, склада, показов, кликов, рекламных расходов, заказов, выручки, себестоимости и остатков [3].

Были сформированы три CSV-набора различного объёма. Это позволило оценить поведение технологий при росте размера входных данных.

Таблица 1 — Структура тестовых данных

Набор данных	Размер CSV	Количество строк
wb_report_10mb.csv	10 МБ	133 461
wb_report_50mb.csv	50 МБ	667 110
wb_report_100mb.csv	100 МБ	1 334 352

Описание аналитического сценария

Для всех технологий выполнялся один и тот же сценарий обработки. Сначала выполнялась загрузка CSV, затем преобразование типов, фильтрация по периоду и брендам, расчёт прибыли и агрегирование данных по дате и бренду. В результате формировались показатели выручки, рекламных расходов, прибыли, количества заказов, кликов, показов, CTR, ROAS и CPO.

Формулы ключевых метрик:

profit = revenue - cost - ad_spend

CTR = clicks / impressions

ROAS = revenue / ad_spend

CPO = ad_spend / orders

total_seconds = t_after - t_before

Каждый вариант запускался три раза, после чего рассчитывались средние значения. Такой подход позволяет снизить влияние случайных колебаний времени выполнения и получить более устойчивую оценку производительности.

Таблица 2 — Результаты измерения времени

Размер CSV	pandas, сек	polars, сек	ClickHouse, сек
10 МБ	0,147	0,037	0,296
50 МБ	0,655	0,049	0,347
100 МБ	1,310	0,093	0,405

По результатам измерений polars показал минимальное среднее время обработки на всех трёх объёмах данных. На файле 100 МБ среднее время выполнения сценария составило около 0,093 секунды для polars, 0,405 секунды для ClickHouse и 1,310 секунды для pandas. При этом ClickHouse в режиме local оказался быстрее pandas на средних и больших объёмах, но уступил polars в данном сценарии, так как запрос выполнялся напрямую по CSV без предварительной загрузки данных в постоянную аналитическую таблицу.

Таблица 3 — Результаты измерения памяти

Размер CSV	pandas, МБ	polars, МБ	ClickHouse, МБ
10 МБ	129,1	91,0	231,5
50 МБ	238,6	190,0	320,0
100 МБ	399,5	315,1	382,5

Сравнение пикового потребления памяти показывает, что polars использовал меньше оперативной памяти, чем pandas, на всех наборах данных. Для файла 100 МБ средний пик памяти составил около 315 МБ для polars и около 399 МБ для pandas. ClickHouse local показал около 382 МБ на 100 МБ CSV, что близко к pandas, но методика измерения отличается: память фиксировалась для отдельного дочернего процесса ClickHouse.

Заключение

В ходе выполнения научно-исследовательской работы был реализован и проведён эксперимент по сравнению трёх технологий обработки больших CSV-отчётов Wildberries-подобной структуры: pandas, polars и ClickHouse local. Для проверки были сформированы тестовые наборы данных объёмом 10, 50 и 100 МБ, содержащие от 133 тысяч до 1,33 миллиона строк.

На основании полученных результатов можно сделать вывод, что polars является наиболее эффективным решением для локальной обработки CSV в рамках выбранного сценария. Библиотека обеспечивает минимальное время выполнения и меньшее потребление памяти по сравнению с pandas. pandas остаётся удобным инструментом для прототипирования и небольших аналитических задач, но при увеличении объёма данных начинает уступать по производительности.

ClickHouse local занимает промежуточное положение в данном эксперименте: он быстрее pandas на больших файлах, но уступает polars при прямом чтении CSV. При этом ClickHouse имеет существенный потенциал для production-сценариев, где данные предварительно загружаются в аналитическое хранилище и затем используются для выполнения множества SQL-запросов, построения витрин и обслуживания нескольких пользователей.

Таким образом, при разработке системы аналитической обработки больших CSV-отчётов, рекомендуется использовать ClickHouse local в качестве основного инструмента обработки данных. Это позволяет обеспечить прямой перенос SQL-запросов в хранилище без переписывания кода, достичь более высокой производительности на тяжелых файлах, а

также заложить архитектурную основу для масштабирования до многопользовательской аналитики и больших объёмов данных при минимальных вычислительных затратах.

Список литературы:

1. Wes McKinney. Data Structures for Statistical Computing in Python // Proceedings of the 9th Python in Science Conference. – 2010. – Pp. 56–61. – DOI 10.25080/Majora-92bf1922-00a.
2. Ritchie Vink. Polars: Blazingly Fast DataFrames in Rust and Python // Proceedings of the Open Source Data Processing Conference. – 2023. – URL: https://pola.rs/.
3. Alexey Milovidov, Maxim Zaitsev. ClickHouse: A Column-Oriented Database Management System for Real-Time Analytics // Proceedings of the VLDB Endowment. – 2020. – Vol. 13, No. 12. – Pp. 3248–3261. – DOI 10.14778/3415478.3415560.
4. Tom Augspurger. Modern Pandas // Proceedings of the Python Data Science Handbook Workshops. – 2018. – URL: https://tomaugspurger.github.io/modern-1-intro.html.
5. Peter Boncz, Marcin Zukowski, Niels Nes. MonetDB/X100: Hyper-Pipelining Query Execution // CIDR 2005: Second Biennial Conference on Innovative Data Systems Research. – 2005. – Pp. 225–237. – URL: http://cidrdb.org/cidr2005/papers/P19.pdf.
6. Jeffrey Dean, Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters // Communications of the ACM. – 2008. – Vol. 51, No. 1. – Pp. 107–113. – DOI 10.1145/1327452.1327492.

References:

1. Wes McKinney. Data Structures for Statistical Computing in Python // Proceedings of the 9th Python in Science Conference. – 2010. – Pp. 56–61. – DOI 10.25080/Majora-92bf1922-00a.
2. Ritchie Vink. Polars: Blazingly Fast DataFrames in Rust and Python // Proceedings of the Open Source Data Processing Conference. – 2023. – URL: https://pola.rs/.
3. Alexey Milovidov, Maxim Zaitsev. ClickHouse: A Column-Oriented Database Management System for Real-Time Analytics // Proceedings of the VLDB Endowment. – 2020. – Vol. 13, No. 12. – Pp. 3248–3261. – DOI 10.14778/3415478.3415560.
4. Tom Augspurger. Modern Pandas // Proceedings of the Python Data Science Handbook Workshops. – 2018. – URL: https://tomaugspurger.github.io/modern-1-intro.html.
5. Peter Boncz, Marcin Zukowski, Niels Nes. MonetDB/X100: Hyper-Pipelining Query Execution // CIDR 2005: Second Biennial Conference on Innovative Data Systems Research. – 2005. – Pp. 225–237. – URL: http://cidrdb.org/cidr2005/papers/P19.pdf.
6. Jeffrey Dean, Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters // Communications of the ACM. – 2008. – Vol. 51, No. 1. – Pp. 107–113. – DOI 10.1145/1327452.1327492.